AUGUST 1991 £3.25
DM. 18

Commodore DISK USER

ANALYSER 64

Get to the bare bones of your C64 programs

THE DISK

Games list creator V2
Ultimate data maker
Basic program maker
Windows C64
File utilities
Labeller C64
X-Ray files
Memory game
Lite menus

9 770953 061014 08

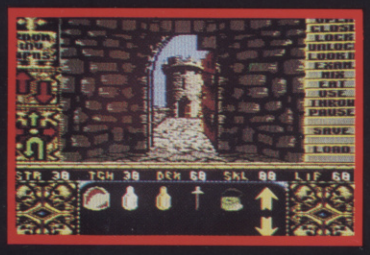
Elvira®

mistress of the dark™

- Hour upon hour of addictive entertainment
- Totally icon-driven for ease of play
- Turbo disk loading guarantees fast graphical access
- An Award Winning game

A SUPERB FANTASY ROLE-PLAYING GAME WITH OUTSTANDING GRAPHICS AND GAMEPLAY

- Over 1 Megabyte of game code supplied on 3 double-sided disks
- Hundreds of locations in and around Elviras Haunted Castle
- Hand to hand real time combat with Knights and Monsters
- Includes instructions and secret spell book for mixing spells and potions



CASTLE RAMPARTS
CBM64

GARDENERS SHED
CBM64

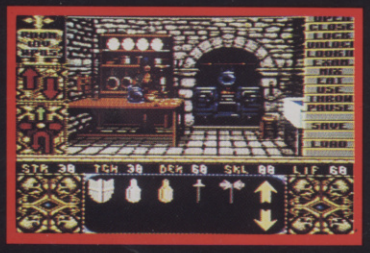
CASTLE ARMOURY
CBM64

AWESOME QUALITY: THE BEST FRP EVER SEEN ON THE CBM 64

ELVIRA rescues you from the Jailers cell and sends you on an unforgettable mission to dispose of the spirit of her great great grandmother **EMELDA.** On your travels in and around the **HAUNTED CASTLE** you will come across some of the most weird and...

FALCON ATTACK
CBM64

....**EVIL** monsters and beings you have ever encountered. From **UNDEAD KNIGHTS** to the **CATACOMB BEAST, WEREWOLVES** to a **VAMPIRE,** and of course wicked **EMELDA** herself, this is the challenge that you will play **AGAIN AND AGAIN!**



ELVIRA IN THE KITCHEN
CBM64

A KNIGHT FIGHTING

ALL SCREENSHOTS
FROM CBM64

THE BLACKSMITHS FORGE
CBM64

For information about the Elvira Fan Club:
14755 Ventura Blvd.,
#1-710, Sherman Oaks, CA 91403, USA

DESIGNED BY HORROR SOFT LTD

FLAIR
S·O·F·T·W·A·R·E

Elvira image © 1990 Queen 'B' Productions. Game design © Horror Soft Ltd. Game and all other materials © Flair Software Ltd. Elvira and Mistress of the Dark are the Trademarks of Queen 'B' Productions.

Available from your local dealer or direct from Flair Software Ltd, The Smithy Side, 7 Belle Villas, Ponteland, Newcastle Upon Tyne. NE20 9BD Priced £24.99

CONTENTS

Commodore DISK USER

Volume 4 Number 10 AUGUST 1991

ON THE DISK

FILE UTILITIES
A collection of Basic programming tools 5

GAMES LIST CREATOR V2
An update of an earlier CDU program 14

ULTIMATE DATA MAKER
Data statements made the easy way 15

LABELLER 64
Create your own labels with ease 18

MEMORY GAME
A computer version of concentration 21

LITE MENUS
An easy way for Basic programmers to produce menus 29

BASIC MAKER
Basic programs produced on a word processor 32

WINDOWS C64
Windowing made easy for all 34

X-RAY FILES
Examine your files in more detail 39

ANALYSER C64
Get to the bare bones of your Basic programs 40

IN THE MAGAZINE

BUG FIX
Monsters and Scroll Writer bugs are corrected 10

BASICS OF BASIC
Our Basic tutorial continues 12

SOFTWARE OFFER
Don't miss out on our games specials 24

TECHNO-INFO
Your favourite guru is back with more answers 26

SACRAMENTO
Is this what the future really holds 30

MACHINE LANGUAGE TECHNIQUES
Learners to MC programming get more help 36

BACK ISSUES
Catch up on your missed issues, back to issue one 44

HACKMAN & SOBBIN 2
Jenni Simpson pours out her heart again 48

Group Editor: Paul Eves
Designer: Mark Newton
Technical Editor: Jason Finch
Program Evaluator: John Simpson
Publishing Consultant: Paul Crowder
Advertisement Manager: Cass Gilroy
Classified Sales: Deborah Curran
Publisher: Hasnain Walji
Distribution: Seymour Press Distribution Ltd. Winsor House, 1270 London Road, Norbury, London SW16 4DH. Tel:081 679 1899. Fax: 081 679 8907
Printed By: Gibbons Barford Print

Subscription Rates

UK	£33.00
Europe	£39.00
Middle East	£39.30
Far East	£41.60
Rest of World	£39.70 or $69.00

Airmail rates on request
Contact: Select Subscriptions. Tel: (0442) 876661

Commodore Disk User is a monthly magazine published on the 3rd Friday of every month. Alphavite Publications Limited, 20, Potters Lane, Kiln Farm, Milton Keynes, MK11 3HF. Telephone: (0908) 569819 FAX: (0908) 260229. For advertising ring (0908) 569819

Opinions expressed in reviews are the opinions of the reviewers and not necessarily those of the magazine. While every effort is made to thoroughly check programs published we cannot be held responsible for any errors that do occur.

The contents of this publication including all articles, designs, drawings and programs and all copyright and other intellectual property rights therein belong to Alphavite Publications Limited. All rights conferred by the law of copyright and other intellectual property rights and by virtue of international copyright conventions are specifically reserved to Alphavite Publications Limited and any reproduction requires the prior written consent of the company

© 1991 ISSN 0953-0614

EDITORS COMMENT

Midsummers day already and not a single suntan to be seen anywhere in the CDU editorial office. What is happening to our weather? The same as what's happening to CDU I suppose. After last months issue, I'm surprised we still have any readers left. What with dropping BASICS OF BASIC, TECHNO-INFO and MACHINE LANGUAGE TECHNIQUES. Then we had yet another reprint of the 1541 article. (It must be popular though considering the number of letters we get asking for it!!) . I think you will all agree though, this months disk is packed with quite a variety of programs to whet your appetities.
The first one "FILE UTILITIES" will be building up over the next three months into quite a comprehensive Basic programmers toolkit. LABELLER64 is ideal for those of you that do your own letter writing and office work. X-RAY FILES (programmed by yours truly) is a boon to those that need to know as much information about a file as is possible. Enough of this waffle, read the contents then load up your drives. We have all systems go!!!!!!

DISK INSTRUCTIONS

Although we do everything possible to ensure that CDU is compatible with all C64 and C128 computers, one point we must make clear is this. The use of 'Fast Loaders', 'Cartridges' or alternative operating systems such as 'Dolphin DOS', may not guarantee that your disk will function properly. If you experience problems and you have one of the above, then we suggest you disable them and use the computer under normal, standard conditions. Getting the programs up and running should not present you with any difficulties, simply put your disk in the drive and enter the command.

LOAD"MENU",8,1

Once the disk menu has loaded you will be able to start any of the programs simply by selecting the desired one from the list. It is possible for some programs to alter the computers memory so that you will not be able to LOAD programs from the menu correctly until you reset the machine. We therefore suggest that you turn your computer off and then on again, before loading each program.

HOW TO COPY CDU FILES

You are welcome to make as many of your own copies of CDU programs as you want, as long as you do not pass them on to other people, or worse, sell them for profit.

For people who want to make legitimate copies, we have provided a very simple machine code file copier. To use it, simply select the item FILE COPIER from the main menu. Instructions are presented on screen.

DISK FAILURE

If for any reason the disk with your copy of CDU will not work on your system then please carefully re-read the operating instructions in the magazine. If you still experience problems then:

1. If you are a subscriber, return it to:
Select Subscriptions Ltd
5, River Park Estate
Berkhamsted
Herts
HP4 1HL Telephone; 0442 876661

2. If you bought it from a newsagents, then return it to:
CDU Replacements
STANLEY PRECISION DATA SYSTEMS LTD
Unit F
Cavendish Courtyard
Sallow Road
Weldon North Industrial Estate
Corby
Northants
NN17 1JX Telephone; 0536 61787

Within eight weeks of publication date disks are replaced free.

After eight weeks a replacement disk can be supplied from STANLEY PRECISION DATA SYSTEMS LTD for a service charge of £1.00. Return the faulty disk with a cheque or postal order made out to STANLEY PRECISION DATA SYSTEMS LTD and clearly state the issue of CDU that you require. No documentation will be supplied.

Please use appropriate packaging, cardboard stiffener at least, when returning disk. Do not send back your magazine, only the disk please.

NOTE: Do not send your disks back to the above address if its a program that does not appear to work. Only if the DISK is faulty. Program faults should be sent to: BUG FINDERS, CDU, Alphavite Publications Ltd, Unit 20, Potters Lane, Kiln Farm, Milton Keynes, MK11 3HF. Thank you.

BASIC FILE UTILITIES

A collection of Basic utilities for the Basic programmer - JOHN CAMPBELL

FILE MENU is the main program in a package of ten utility programs (known collectively as the FILE UTILITIES), which give you the capability to manipulate Basic program files on disk. These utilities are a vital toolset for you, the Basic programmer! Here is a peek at what you will be getting:

1. FILE MENU - This utility is the main menu for the FILE UTILITIES. From it, you can run any of the other utilities, and you can even load your own programs from the menu. In addition, FILE MENU allows you to print or display a directory of the disk, selectively delete files from the disk, and perform any disk command.

2. FILE RENUMBER - This utility takes a Basic program file on disk and creates a new version of the file which is completely renumbered. As well as changing the line numbers themselves, FILE RENUMBER automatically changes the GOSUBs and GOTOs to the correct new line number, unlike some renumbering programs.

3. FILE EXTRACTOR - This utility extracts a range of lines from a Basic program file on disk and creates a new file with those lines in it. Thus a handy subroutine can be extracted from one program for use in another.

4. FILE DELETER - This utility deletes a range of lines from a Basic program file on disk and creates a new file with those lines removed from it. This is the complement of FILE EXTRACTOR.

5. FILE MERGER - This utility merges two Basic program files into a single new file. The resultant file contains the lines from both input files merged in numerical order. This is how you merge your handy subroutine into your new program, after you have renumbered it.

6. FILE SEARCHER - This utility searches one or more Basic program files for a character string, and prints it on the screen or printer. It allows you to preview all occurrences of a string in a file before you replace it with a new string.

7. FILE REPLACER - This utility searches a Basic program file for a character string, and replaces every instance of it with another string. It allows you to change a popular variable name in your handy subroutine so it does not conflict with your new program, or change all the PRINT statements to PRINT# statements.

8. FILE DIFFER - This utility compares two Basic program files and displays or prints the differences. It allows you to pinpoint exactly what you changed in your current version of the program which makes it run so much worse than the last version.

9. FILE CROSSREF - This utility produces a cross reference of all variables in a Basic program and the lines in which they occur. It does the same thing for subroutines (GOSUBs). It allows you to identify and correct misspelled variables, variables set but never used, and variables used but never set.

10. FILE LISTER - This utility lists a Basic program file to the screen or printer without having to leave FILE UTILITIES or open a channel to the printer. It allows you to list your newly merged file to make sure your handy subroutine ended up in the right place in the file.

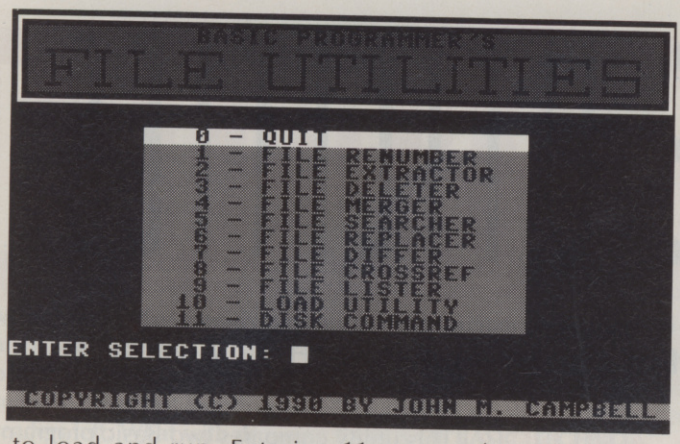
When you run FILE MENU, the menu of utilities is displayed, from which you may select the utility you wish to run. (Of course, you may also load and run any of the utilities directly.)

```
FILE UTILITIES MENU

 0 -  RETURN TO BASIC
 1 -  FILE RENUMBER
 2 -  FILE EXTRACTOR
 3 -  FILE DELETER
 4 -  FILE MERGER
 5 -  FILE SEARCHER
 6 -  FILE REPLACER
 7 -  FILE DIFFER
 8 -  FILE CROSSREF
 9 -  FILE LISTER
10 -  LOAD UTILITY
11 -  DISK COMMANDS
```

ENTER SELECTION?

Entering a zero, or simply a RETURN, will end the program and return to Commodore Basic. Entering 1 - 9 loads and runs the indicated utility program. Entering 10 causes the program to prompt you for the utility you want



to load and run. Entering 11 causes the program to prompt you for a disk command.
You may also make a menu selection with the joystick or with the cursor keys. With the joystick connected to Control Port 2, you simply push it forward or pull it backward to position the highlight bar over the menu item you desire. Then press the fire button to select that item. Similarly, you may use the up-arrow and down-arrow cursor keys to move the highlight bar, and press the RETURN key to make your selection.

When you load and run a File Utility, you are prompted to enter one or more input files. Once you have entered the input file name(s), you are prompted to select where the output is to go. In some cases, that is an output file name. In other cases, you select output to be displayed on the screen or printed on the printer. Any additional information required by the utility program is requested separately (e.g., the range of program lines to be operated on).

When the utility is finished you see this prompt:

LOAD MENU, QUIT, OR REPEAT (M,Q,R)?

Typing "M" loads the main menu program again as long as it exists on the disk. Typing "Q" quits the program and returns to Basic. Finally, typing "R" repeats the program again. The Basic program lines which perform these tasks are provided for you in a file called LDMENU.BAS. Use the FILE MERGER program to merge it to the end of your own utility, so it will return to FILE MENU when it finishes. Finish the merger by loading the resultant file and performing these steps:

a. Change the END statement in your utility to a GOTO 10000.

b. Change line 10040 to go to the first line in your utility program.

When you select the disk command from the menu, the program asks you for the command you wish to enter. Any of the valid disk commands are accepted, including the disk directory. You have the convenience of being able to do your disk maintenance within the program, without needing to stop and exit the program, then reload it when you want to use another utility. The following is a list of the disk commands:

CMD	DESCRIPTION
$	DISK DIRECTORY
@	READ ERROR CHANNEL
C	COPY FILE
I	INITIALISE DRIVE
N	FORMAT NEW DISK
R	RENAME FILE
S	SCRATCH FILE
V	VALIDATE DISK

See your disk drive manual for a full description of the command formats.
The directory command ($) displays the directory of the current disk to the screen, or prints it on the printer. It allows you a convenient way to print out the directory of each disk, which you can keep tucked away in the sleeve of the disk for handy reference. You will also want to print the disk directory before proceeding to to delete old disk files.

Note that FILE MENU supports the wildcard specification in both the disk directory and scratch file commands. In addition, FILE MENU prompts for confirmation of the destructive commands, scratch file and format new disk, to allow just a bit more protection of your files, by giving you the chance to abort the command. That helps you avoid deleting files by accident, as well as whole disks.

When you specify a wildcard in the scratch a file command (S), FILE MENU searches the disk directory and displays in turn each file name which fits the criteria, asking for confirmation. If you indeed wish to delete the file, answer the prompt with Y (for Yes) and RETURN. If you do not want to delete the file, answer N (for No) and RETURN. If you wish to cancel the scratch command, enter a Q (for Quit) and RETURN, and the program will return you to the main menu.

When you specify the format new disk command (N), File Menu asks "ARE YOU SURE?" If you wish to continue with the format command, answer Y (for Yes) and RETURN. Then the program instructs you to insert the disk to be formatted and type any key to start the formatting process. Make sure the disk in the drive is the one you want to format before pressing a key. If you wish to cancel the format command, answer N (for No) and RETURN. The program will return you to the main menu.

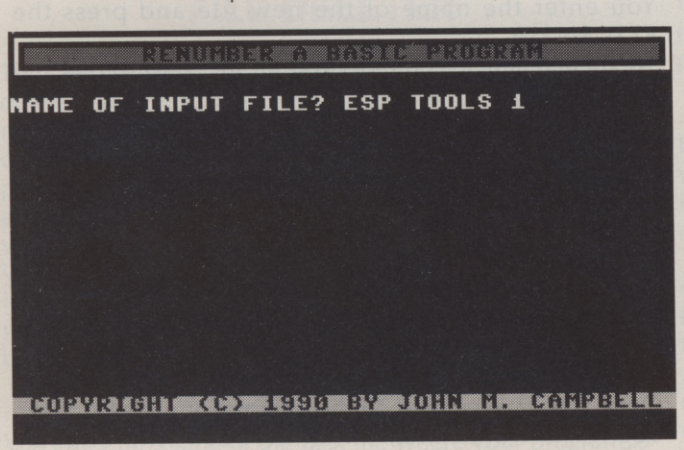
You may switch disks if you want to perform a disk command on a different disk, but you must do so before typing in the disk command. The program will prompt you to switch the FILE MENU disk back into the drive before completing the command. To avoid having to switch disks for every new command, you can copy FILE.VER from the FILE MENU disk to your working disk. This file serves as a reference point to which File Menu returns after each command. You can use any Copy program to copy FILE.VER to your own disks.

AS MENTIONED AT THE START OF THIS ARTICLE, WE WILL BE BRINGING YOU THE ENTIRE SET OF UTILITIES OVER THE NEXT 3 ISSUES OF THE MAGAZINE. THIS MONTH WE GIVE YOU FILE

RENUMBER, FILE EXTRACTOR AND FILE DELETER. EACH MONTH YOU WILL GET THE FILE MENU PROGRAM SO THAT YOU CAN COPY THE UTILITY FILES AS WE PUBLISH THEM. EVENTUALLY YOU WILL HAVE A SEPARATE DISK WITH ALL THE FILE UTILITIES ON IT.

FILE RENUMBER

The File Renumber program is the first utility a Basic programmer needs. Even the most careful programmers who leave plenty of room between line numbers will face at some point the problem of needing to insert more lines than there is space available. The only choice is to



renumber the lines in the file, at least partially. Doing it by hand is labor-intensive and error-prone at best, but File Renumber makes the task automatic and error-free. You just select the line number you want to start at and the increment to use between lines, and the utility creates a new file with the proper line numbers. Furthermore, File Renumber searches out the GOTOs, the GOSUBs, the IF-THENs, and the RUN statements in your program and replaces the old line number references with the new line numbers corresponding to the same Basic statements as the original program.

When you select the File Renumber utility from the Menu, it is loaded and run. The utility first asks you to supply the name of the existing Basic file to be renumbered:

NAME OF INPUT FILE?

You enter the name of the disk file where your program is stored and press the RETURN key.

Next, the utility asks you to supply the name you want to use for the disk file to be created to store the renumbered program:

NAME OF RENUMBERED FILE?

You enter the name of the new file and press the RETURN key.

Then File Renumber prompts you for the starting line number to use in the new program:

STARTING LINE NUMBER?

You enter the number you want used for the first line in the file, which might also be the line where the subroutine is to start, if the file will be merged with another program later.

Finally, the utility prompts you for the line increment to use:

LINE NUMBER INCREMENT?

You enter the amount to add to each line number to get the next line number. This number is usually 10, but you may make it bigger to allow for inserting more than ten lines, or you may make it smaller to allow you to merge the lines into a smaller space in another program.

Once you have entered all the information, File Renumber begins its work. First, the utility must read through the original program file and catalog all the existing line numbers. As it finds each line, the utility updates the display screen to let you know how many lines it has read. When it reaches the end of the file, File Renumber starts reading the original file again from the beginning.

During this second pass through the file, the utility copies each line to the new file, changing line numbers as it goes. As it reads each line from the old file, File Renumber checks for the types of Basic statements which refer to other line numbers in the program. When it detects a GOSUB or a GOTO, an ON-GOSUB or an ON-GOTO, an IF-THEN or a RUN, the utility reads the line number which follows. It looks up that line number in the list made in the first pass through the file, then substitutes the new line number into the statement it writes to the new file. Each time a line is written to the new file, the display screen is updated accordingly. Once the second pass through the file is completed, the renumbered program exists on the disk.

There are five possible error messages you may get from File Renumber.

1. ERROR—FILE NOT FOUND

File Renumber could not find the original file you want to renumber. You need to check the spelling of the file name, and make sure that file is on the disk. Then run the program again with the correct file name.

2. ERROR—FILE EXISTS

File Renumber found a file already existing with the name you want to use for the renumbered file. File Renumber cannot replace an existing file. You need to check the spelling of the file name and either delete the existing file or use a different file name for the renumbered file. Then run the program again with the correct file name.

3. ERROR—LINE NUMBER TOO LARGE

A Basic line number must be within the range of 0-

```
EXTRACT LINES FROM A PROGRAM

NAME OF INPUT FILE? FILE MENU
NAME OF OUTPUT FILE? FM1
STARTING LINE NUMBER? 10
ENDING LINE NUMBER? 34

COPYRIGHT (C) 1990 BY JOHN M. CAMPBELL
```

63999. During the renumbering process, the maximum line number was exceeded. You need to either reduce the starting number, reduce the increment, or both. You can use the number of Basic lines which File Renumber prints out to calculate if your starting line and increment is going to exceed the maximum line number. No new file is created if this error occurs. Run the program again with new starting line number and increment.

4. ERROR—TOO MANY LINES IN FILE

This error should never occur. It means there are more than 6000 lines in a Basic program. However, since a line of Basic takes at least 8 bytes (usually much more) there is not enough free memory on the Commodore 64 to exceed 6000 lines. No new file is created if this error occurs.

5. ERROR—FILE MENU NOT FOUND

This error occurs when you elect to load the File Menu after completing execution of the utility, but it is not found on the disk. You are prompted again to enter your choice, which gives you the opportunity to insert the proper disk into the drive before responding.

FILE EXTRACTOR

The File Extractor program allows the Basic programmer to extract a range of lines from a program file and create a new file containing only those lines extracted. This

```
EXTRACT LINES FROM A PROGRAM

NAME OF INPUT FILE? FILE MENU
NAME OF OUTPUT FILE? FM1
STARTING LINE NUMBER? 10
ENDING LINE NUMBER? 34
SEARCHING FOR STARTING LINE
NUMBER OF LINES FOUND=
NUMBER OF LINES OUTPUT=
ERROR--NO LINES FOUND WITHIN RANGE

LOAD MENU, QUIT, OR REPEAT (M,Q,R)?

COPYRIGHT (C) 1990 BY JOHN M. CAMPBELL
```

capability is especially useful when extracting a subroutine, either to use in another program, or when rearranging lines in a program. Once extracted, you may use the File Renumber utility to renumber the lines and the File Merger utility to merge them into another program.

When you select the File Extractor utility from the Menu, it is loaded and run. The utility first asks you to supply the name of the existing Basic file from which the lines are to be extracted:

NAME OF INPUT FILE?

You enter the name of the disk file where your program is stored and press the RETURN key.

Next, the utility asks you to supply the name you want to use for the disk file to be created to store the extracted program lines:

NAME OF OUTPUT FILE?

You enter the name of the new file and press the RETURN key.

Then File Extractor prompts you for the range of lines to be extracted:

STARTING LINE NUMBER?

You enter the first line number in the range of lines to be extracted.

ENDING LINE NUMBER?

You enter the number of the last line to be extracted from your program. Of course, the ending line number must be equal to or larger than the starting line number.

Once you have entered all the information, File Extractor begins its work. First, the utility reads the original program file, searching for the starting line you specified. As it finds each line, File Extractor updates the display screen to let you know how many lines it has found.

When the starting line is encountered, it is copied into the new file with the name you indicated. As each subsequent line is copied into the new file, the utility updates the display screen to show how many lines have been output. The processing ends when either the ending line is detected or the end-of-file is encountered in the original file.

There are five possible error messages you may get from File Extractor:

1. ERROR—FILE NOT FOUND

File Extractor could not find the original file from which you want to extract lines. You need to check the spelling of the file name, and make sure that file is on the disk. Then run the program again with the correct file name.

2. ERROR—FILE EXISTS

File Extractor found a file already existing with the name you want to use for the new file. File Extractor cannot replace an existing file. You need to check the spelling of the file name and either delete the existing file or use a different file name for the output file. Then run the program again with the correct file name.

3. ERROR—NO LINES FOUND WITHIN RANGE

File Extractor found no lines to extract from the original file within the range of line numbers specified. No output file is created. List the file again to determine the proper range of lines, then run the program again.

4. STARTING LINE MUST BE < ENDING LINE

File Extractor detected a mistake made in specifying the range of lines to be extracted. The utility displays this message then allows you to re-specify the line number range by prompting you for the starting and ending line numbers again.

5. ERROR—FILE MENU NOT FOUND

This error occurs when you elect to load the File Menu after completing execution of the utility, but it is not found on the disk. You are prompted again to enter your choice, which gives you the opportunity to insert the proper disk into the drive before responding.

FILE DELETER

The File Deleter program allows the Basic programmer to delete a range of lines from a program file and create a new file containing only those lines remaining. This capability is often useful when preparing to merge two programs together. You simply delete the unwanted lines and use the File Merger utility to merge the resulting file with another program.

When you select the File Deleter utility from the Menu, it is loaded and run. The utility first asks you to supply the name of the existing Basic file from which the lines are to be deleted:

NAME OF INPUT FILE?

You enter the name of the disk file where your program is stored and press the RETURN key.

Next, the utility asks you to supply the name you want to use for the disk file to be created to store the program lines remaining after deleting the unwanted lines:

NAME OF OUTPUT FILE?

You enter the name of the new file and press the RETURN key.

Then File Deleter prompts you for the range of lines to be deleted:

STARTING LINE NUMBER?

You enter the first line number in the range of lines to be deleted.

ENDING LINE NUMBER?

You enter the number of the last line to be deleted from your program. Of course, the ending line number must be equal to or larger than the starting line number.

Once you have entered all the information, File Deleter begins its work. First, the utility reads the original program file, searching for the starting line you specified. As it finds each line, File Deleter copies it into the new file you indicated and updates the display screen to let you know how many lines it has output.

When the starting line is encountered, it is ignored, and the utility updates the display screen to show how many lines have been deleted. The line number is checked for each successive line read from the original file until the ending line is detected. At that point, File Deleter switches back to copying all remaining lines from the original file to the output file. Again, the display screen is updated to reflect each line output. Processing ends when the end-of-file is encountered in the original file.

There are five possible error messages you may get from File Deleter:

1. ERROR—FILE NOT FOUND

File Deleter could not find the original file from which you want to delete lines. You need to check the spelling of the file name, and make sure that file is on the disk. Then run the program again with the correct file name.

2. ERROR—FILE EXISTS

File Deleter found a file already existing with the name you want to use for the new file. File Deleter cannot replace an existing file. You need to check the spelling of the file name and either delete the existing file or use a different file name for the output file. Then run the program again with the correct file name.

3. ERROR—NO LINES FOUND WITHIN RANGE

File Deleter found no lines to delete from the original file within the range of line numbers specified. No output file is created. List the file again to determine the proper range of lines, then run the program again.

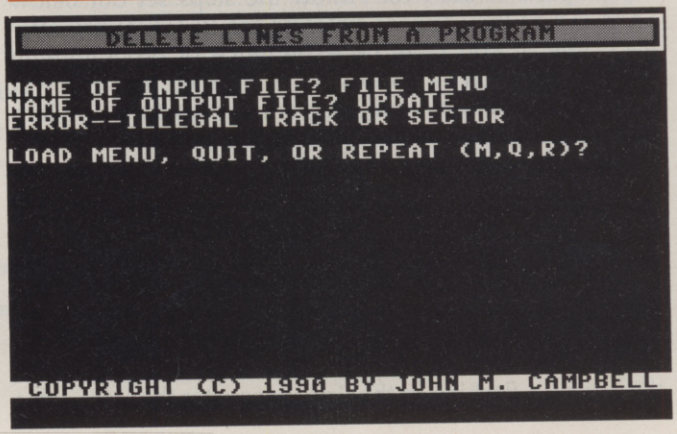
4. STARTING LINE MUST BE < ENDING LINE

File Deleter detected a mistake made in specifying the range of lines to be deleted. The utility displays this message then allows you to re-specify the line number range by prompting you for the starting and ending line numbers again.

5. ERROR—FILE MENU NOT FOUND

This error occurs when you elect to load the File Menu after completing execution of the utility, but it is not found on the disk. You are prompted again to enter your choice, which gives you the opportunity to insert the proper disk into the drive before responding.

That just about wraps up this months offering, next month we will be giving you FILE MERGER, FILE SEARCHER and FILE REPLACER. Until then, experiment with the three files on this disk!!


DELETE LINES FROM A PROGRAM
NAME OF INPUT FILE? FILE MENU
NAME OF OUTPUT FILE? UPDATE
ERROR--ILLEGAL TRACK OR SECTOR
LOAD MENU, QUIT, OR REPEAT (M,Q,R)?
COPYRIGHT (C) 1990 BY JOHN M. CAMPBELL

EDITORS EXTRA BIT

Jason Finch comes to the Rescue!

We have received a number of letters from readers concerning the utility program SCROLL WRITER and the game MONSTERS, both featured on the June 1991 disk. With the former, the problem is actually viewing the text that has been entered, and with the latter the problem is how to get the cursor to appear on the menus. There is a solution to the problems but I will let Jason, our resident technical guru explain what is causing the problems and how they can be rectified...

THE PROBLEMS

Well, the problem is caused by the different ways in which the older models of C64 (pre-1985) and the newer ones (and C128s in 64 mode) clear the screen. Each time the screen is cleared, all the colour memory (55296-56295) is restored to a particular value. When using an old C64 the colour that is stored in each colour memory "slot" is that of the background colour when the screen was cleared. But with the newer models, the colours are set to that of the cursor at the time. Should you be unfortunate enough to own one of the archaic C64s of the late seventies (rare!) then you may have the problem that none of the colour memory is reset at all! As an example, upon power-up, the background colour is blue and the cursor colour is light blue. If you clear the screen and enter POKE 2023,65 one of two things will occur. Either: A light blue "A" will appear in the bottom left corner or, if you own an older model, nothing will happen. You will have to enter POKE 56295,14 to give that "A" its colour. It is there anyway but the colour is that of the background so you don't see it. I hope that all of you have at least understood the reason for the problems. Now, if you own a newer model of the C64 and are not experiencing any problems then lucky you - skip this feature because it will possibly make matters worse for you if you carry out the changes for people with older C64s!

MONSTERS FIX

Ok, firstly how to deal with MONSTERS because it is the simplest of the two! Follow the steps set out below EXACTLY as they appear, pressing the RETURN key after each line. If you have no experience of programming then do not worry - do exactly as detailed and you will be able to convert your program successfully! Switch your computer off, wait a few seconds and switch it back on. Then do the following:

LOAD "MONSTERCODE",8,1
SYS 5095: POKE 808,237: LIST 135

Now, a BASIC line will be displayed. Use the cursor keys to position the cursor over the "5" in the line number - 135. Press the number six. Now press the cursor down key once and then the cursor right key until the cursor is flashing in the first space after the line - after the final quote and the letters TU (not of the word TURN). Now press the COLON (:) key and type the word RETURN. Now press the RETURN key. Now continue typing in the following entries, pressing RETURN after each:

135 POKE 53281,2: PRINT CHR$(147): POKE 53281,10
POKE 43,180: POKE 44,19
SAVE "@0:MONSTERCODE",8

SCROLL WRITER FIX

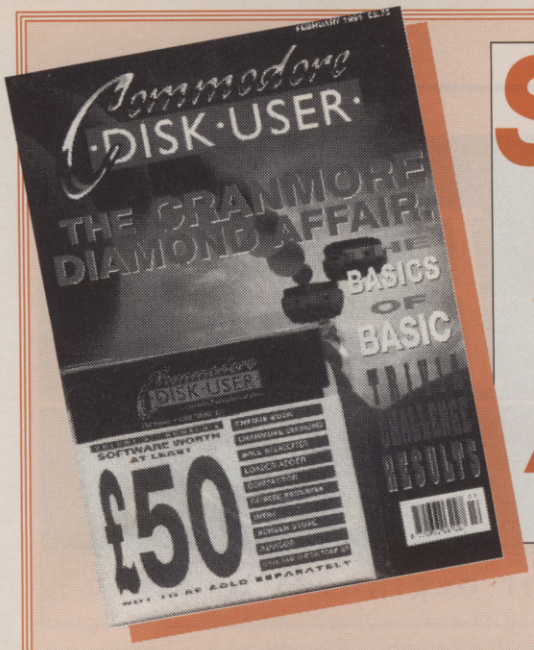
There, that was pretty simple wasn't it? Now, to rectify the problem with SCROLL WRITER you will need a basic understanding of machine language together with a plug-in cartridge like ACTION REPLAY or SUPER SNAPSHOT that allows you to change a frozen program and then restart it (note that this is illegal on commercial software - you may alter this program as you have the permission of the copyright holders). If you do not own such a device then you will not be able to correct the problem yourself. Send your disk to CDU Techno-Info at 11 Cook Close, Rugby, Warwickshire, CV21 1NG for conversion of both programs - and include at least 33p worth of stamps if you'd like it returned! For those that possess a freezer cartridge, read on...

LOAD and RUN the program "06-WRITER*" from the disk. When the main menu appears press the "Freeze" button or equivalent on your cartridge. Now do whatever you need to in order to enter the MONITOR on the cartridge. Now you must enter the following lines. Check the format required in the manual and alter the commands slightly if necessary:

T 0858 087D 085B (this command should transfer memory $0858 to $087D incl. to $085B)
A 0858 JSR $E544
A 0852 LDA $0824
A 087A LDA $0820
A 087D STA $D021
X (to return to cartridges freeze menu)

Now restart the program - when you test your demo or create one of your own it will be saved with a corrected piece of code installed. This code will not alter your chosen colours, it just does things a different way around so that the screen is cleared to the correct colour. With a bit of luck, some of you will have been able to carry out that "repair". Once again, if you cannot for whatever reason, please send your disk to CDU Techno-Info for a free replacement - remembering to enclose 33p to cover return postage.

I have tried, as usual, to explain everything in a very simple to understand manner so that both the beginner and experienced will be able to alter the programs without too many hassles. This is your friendly neighbourhood guru signing off for now. See yer!



Subscribe now . . . And Save £9

OR KICK YOURSELF FOR THE REST OF THE YEAR . . .

We've gone mad and are offering you the opportunity of receiving Commodore Disk User's next twelve issues for the staggeringly low price of *£30 if you live in the U.K. We will even post it to you free as well.**

The current price of Commodore disk user for 12 issues is £39,
however if you are smart you can save £9 by subscribing to this offer, so don't delay the offer ends June 28th 1991
CDU is the only magazine worth considering in the world of the serious C64 Commodore user.

Published monthly –

COMMODORE DISK USER is the answer to every Commodore computer owner's dream. The disk supplied with the magazine contains a variety of ready to use, high quality computer programs – no more lengthy typing in of listings. The scope of the programs is wide, varying from games to business software and high-powered disk utilities – and the disk would retail for at least £50.00 if bought independently.

Of course, that isn't all. The magazine, besides containing full and comprehensive instructions for using the disk, is a complete computer journal in its own right, with news, reviews, programming, competitions and general interest features.

PRIORITY ORDER FORM

Please commence my subscription to Commodore Disk User with the .....issue.

I enclose a cheque/postal order for £..... made payable to **ALPHAVITE PUBLICATIONS LTD.**

or debit £.....from my Access/Visa Card No:

Valid from.....to.....

Signature.....Name.....

Address.....

.....Post code .....

Cut out and send this form with your remittance to: **SELECT SUBSCRIPTIONS LTD.**
5 RIVER PARK ESTATE, BERKHAMSTEAD, HERTS HP41ML

.....Offer ends JUNE 28th 1991.....

* Rates refer to subscriptions sent post free to UK addresses. Overseas rates on request.

BASICS OF BASIC

BASICS OF BASIC

John Simpson continues his series for beginners to Basic

We are sorry we could not bring you BASICS OF BASIC last month. This was due to various technical difficulties and last minute editorial changes. Up to date, we have covered quite a lot of ground in this introduction to Basic programming. This month sees the commencement of PART SIX which deals mainly with CURSOR TRICKS.

In Part Four we looked at cursor positioning where I demonstrated the use of the TAB and SPC functions. I mentioned then that there are other ways to shift the cursor around the screen, and one method in particular which we shall now examine is to let the Computer's Operating System to do it for us.

THE PLOT DEEPENS

The routine is called PLOT and it commences from memory location 58640. This is NOT a line number but an address within the 64's ROM memory. We cannot 'call' this routine like we do our own subroutines using the keyword 'GOSUB'. Instead we must use a method which calls Machine Language (ML) programs, namely 'SYS'. (This is the most common way to mix a Basic program with a ML program.) The ML program starts at the location given by the SYS statement, in our case 58640. So, a Basic line, such as:

```
100 SYS 58640
```

will divert program control to the ML program which starts at memory address 58640. At the end of the ML program there will be a command (RTS) which means the same as RETURN, and from there the program control is returned back to the instruction following the SYS 58640.

A SLIGHT DIVERSION FROM PLOT

There are several methods of program development, and one of these is known as structured programming which we shall gradually adopt during this series. In structured programming it is common practise to construct a routine which can be called SETUP, and it is from within this routine where we 'set up' and initialise variables, constants, the start screen and colours, dimension and fill arrays, among other things. It is here where we would could initialise memory locations to constant labels, such as:

```
50000 REM **** PROGRAM SETUP ****
50010 REM
50020 PLOT=58640:CR=214:CC=211:REM PLOTTING
CONSTANTS
50030 BO=53280:PA=53281:REM BORDER
/SCREEN(PAPER) CONSTANTS
50040 etc. etc.
.....
.....
59999 RETURN
```

BACK TO PLOT

To use the PLOT routine we must know where we want the cursor to relocate on the screen, and if we want the cursor to return back to its original location, then we need to know from where it came. The cursor's Row and Column positions are held in memory locations 214 and 211 respectively. These are RAM locations, outside of the Basic area, which are constantly updated by the system as the cursor moves around the screen. However, because they are located in RAM we can write (poke) and read (peek) to and from them - effectively changing them. In the SETUP example above I labelled them as CR (Cursor Row), and CC (Cursor Column).
From all of this it can be seen that two routines are needed to exploit the potential of PLOT. (1) a routine to peek and save the current location, and to move the cursor to the new location, and (2) another to poke back the old location values to move the cursor to it's original position.

```
5000 REM *** NEW PLOT ***
5010 R2 = PEEK(CR): C2 = PEEK(CC)
5020 POKE CR,R1:POKE CC,C1
5030 SYS PLOT
5040 RETURN
5050 :
5060 REM  *** RESTORE PLOT ***
5070 POKE CR,R2:POKE CC,C2
5080 SYS PLOT
5090 RETURN
```

As you now know the screen is divided into a matrix of 0-39 columns (across the screen), and 0-24 rows (down the screen). If you need to shift the cursor then allocate to R1 and C1 the new positions you require, call NEW PLOT, and when control returns do whatever it is you want to do at the new screen position, then finish by calling RESTORE PLOT. Let us examine a typical use.

We need to print a message at the foot of the screen, such as "HIT ANY KEY". A typical program segment may look like this:

```
80 .....
90 (BEFORE CURSOR MOVE)
100 R1=24:C1=11:GOSUB 5000     :REM NEW PLOT
110 PRINT "< STRIKE ANY KEY >"
120 GOSUB 5060                 :REM RESTORE PLOT
130 GET A$: IF A$=""THEN130
140 (REST OF PROGRAM)
```

In this example we shifted the cursor to the bottom row of the screen (line 100) and printed our requester roughly halfway across the row (line 110), then returned the cursor back to where it came from (line 120) to wait for the user's response (line 130).

CURSOR STRINGS

Another method to move the cursor around the screen is to create row and column cursor strings. Remembering our mnemonics, or abbreviations, from Part Four,

[CDn] = CURSOR UP/DOWN KEY 'n' NUMBER OF TIMES, and [CRn] = CURSOR LEFT/RIGHT KEY 'n' NUMBER OF TIMES.

First, during SETUP initialise two strings thus:

```
.... ROW$="[HOME][CD25]": REM TO MOVE THE CURSOR DOWN
.... COL$="[CR40]"       : REM TO MOVE THE CURSOR RIGHT
```

When we need to position the cursor anywhere on the screen we simply cut the required string to the length needed! Here is an example.

```
10 ROW$="[HOME][CD25]"
20 COL$="[CR40]"

100 PRINT"[CLR]PAGE ONE..."
110 PRINT"[CD]A GOOD METHOD OF FORMATTING SCREEN INFORMATION IS TO USE STRING ";
120 PRINT"MANIPULATION. IN OTHER WORDS, THE CUTTING UP OF CURSORSTRINGS."
130 GOSUB 1000
140 PRINT"[CLR]PAGE TWO..."
150 PRINT"[CD]SOMETIMES THIS METHOD IS NOT AS GOOD AS THE PLOT METHOD BUT ";
160 PRINT"EACH HAVE THEIR ADVANTAGES."
170 GOSUB 1000
180 PRINT"[CLR]PAGE THREE...":GOSUB 1000: END

1000 C$=LEFT$(COL$,15):R$=LEFT$(ROW$,23)
1010 PRINT R$ C$ "ANY KEY FOR NEXT PAGE >>"
1020 GETA$:IF A$=""THEN1020
1030 RETURN
```

In the example I have used a situation of printing pages of information to the screen for the user to read. Once the desired page length is reached, then a call to a subroutine is executed which will always print at the same screen location a requester and wait for an 'any key' response before returning to clear the screen and start printing the next page.

Looking more closely at the subroutine we can see that on line 1000 we have employed the LEFT$ method of cutting ROW$ and COL$ to the required length and storing them in R$ and C$. Line 1010 then relocates to the new cursor position before it prints "ANY KEY FOR NEXT PAGE >>". Before it repositions the the cursor it HOMES it. We need to do this first otherwise it would count down and right from whatever position it happened to be in. All that is required is to ensure that of the two cursor strings, the first one to be executed will be the one with the [HOME] command embedded within it.

You can experiment with both methods of cursor manipulation. You will quite easily get the hang of each method.

TO BLINK OR NOT TO BLINK

Before we leave the cursor I thought you might like to know that we can also control whether or not the cursor is on or off.

The cursor always blinks on and off when the 64 is expecting input, that is why when you use the INPUT command there is the cursor, blinking away quite happily. However, when you input with the 'GET' command the cursor is disabled. Which is okay because most of the time we don't want the blinking thing, however, there are times when it would be nice to use the cursor during a GET event sequence.

There is another ROM memory location, 204 this time, which acts as a flag for the cursor. If this location contains any value other than zero then the computer understands this to mean that the cursor is off. A value of zero, naturally, has the opposite effect. We can make use of this, for example, when before a GET command we turn the cursor on, then turn it back off after a key press:

```
100   POKE 204,0
110   GETA$:IFA$=""THEN 110
120   POKE 204,1
130   PRINT A$
140    END
```

Line 100 turns the cursor on, and it will continue blinking until a key is pressed from Line 110. Line 120 turns the cursor off, and line 130 prints which key was pressed.

A problem can occur! It may happen that a cursor which is turned off during its 'on' phase will leave a reversed space (or a coloured square) upon the screen. This can be avoided by use of another memory location that indicates which phase the cursor is in. All we need to do is read this location and wait, if necessary, until the cursor is off. The memory location for this is 207, and the line of code to check this is:

```
115 IF PEEK(207) THEN 115: REM WAIT UNTIL THE CURSOR BLINK IS OFF.
```

Of course you can link the plot routines with this to get the cursor to blink anywhere on the screen you like. That's all for this month I'm afraid. Next month we will be looking at SORTS and SEARCHES, until then, have a good month and keep practicing.

GAMES LIST CREATOR

Keep a record of all your games disks with this UPDATED versatile and novel utility (previously published in AUGUST 90) - JOHN KAY

Rather than make up some suitable text to justify the re-printing of this utility, I present you with part of the original letter sent to the editorial office. I think this describes the changes much better than I could.....

Dear CDU,
Please find an update to GAMES LIST CREATOR published last AUGUST. This program now contains a doctored file which now enables a new Basic program (named 'C') to be loaded and run. I felt the original program was lacking in that (especially NEW LIST ASSEMBLY) one would not be able to see the file names to type in.....

GAMES LIST CREATOR is a utility that enables you to keep a record of all your games disks, which when run will display your lists in a pleasing and musical manner. To use the program is simplicity itself as no knowledge of machine code programming is necessary. First of all take a blank formatted disk and copy the program GAMELIST CREATOR from the CDU disk. Load and run the GAMELIST CREATOR program which will then present you with the main menu screen. There are four options on the menu which are;

1. CREATE A NEW LIST
2. ADD TO AN OLD LIST
3. CREATE A NEW SCROLLING MESSAGE
4. RUN GAMES LIST

Option 1 is the first one that you will have to use. {Options 2 and 4 will not work unless you have already use option 1}.

CREATE A NEW LIST

First of all to create a new list take option 1. When activated you will be displayed a message telling you to type "Q" when finished. Remember this. Press any key to continue with your choice. Displayed in the top left corner is an asterisk, this is now your cursor. Below is a message asking you to enter the title of a game, then a message saying how many games you have entered and finally a message informing you of the option you are in. To enter a game title simply type the name and press return to store the name in memory. Repeat this operation until all your titles are entered. Once finished, type "Q" and you are then ready to save your list to disk for future recall. First of all, any old "Games Lists" are wiped and your new list is saved to disk.

ADD TO OLD LIST

Consider you have now just purchased a few more new games and you want to add them to your list. Simply load and run your newest Games List program and select option 2 from the main menu. Entering the new names is the same as for option 1. Please note that whilst using options 1 and 2 any characters except # and $ can be used. A game title can be up to 29 characters long. The program allows for 14K for the list, which should be sufficient for over 1000 titles.

CREATE NEW SCROLLING MESSAGE

In the bottom border the program incorporates a scrolling message. What this message is, is entirely up to you. When you take this option, you will first be asked the question Size of space (1-5)? This means how much space you want between each word on the message. It is recommended that 2 or 3 is chosen. You will now see the asterisk at the top left of the screen and the message "End here", at the bottom left. Type in your message using the keys A-Z only. Do not press the return key until you have finished typing in your message. Do not worry about the words splitting or appearing to be joined up. The program incorporates a word wrap facility. Once the program has read your message and stored it on disk, press any key to continue.

RUN GAMES LIST

Now for the big one, your list is up to date, you have created a nice witty message and you now want to see it all in action. Option 4 of the main menu will enable you to do this. The program will display a list of 2 files which will needed to be loaded, press any key to start the load. When they have finished loading any key will run the games list. When run you will be faced with a static screen, this is for people with cartridges that wish to freeze out the program. The static screen drops a little hint that it is waiting for you to press fire on your joystick. So insert your joystick into Port 2 and press fire.

GAMES LIST CREATOR
WRITTEN BY J.KAY
MENU
[1] CREATE A NEW LIST.
[2] ADD TO AN OLD LIST.
[3] CREATE A NEW SCROLLING MESSAGE.
[4] RUN GAMES LIST.
COPYRIGHT COMMODORE DISK USER 1990

ULTIMATE DATAMAKER

For those of you that still wish to convert M/Code into Basic DATA Statement, I present the ULTIMATE DATAMAKER - MAHMOOD MERCHANT

Various programs have been written in the past to convert Machine Code or Data Bytes into Basic DATA STATEMENTS. Those written in Basic often used the dynamic keyboard technique to add the lines and were quite slow especially where a large number of statements were to be created. The solution was to resort to Machine Code DATA generators. Most of these required the user to specify a range of memory and the statements were quickly generated. Some others even generated the loader required to re-POKE the data in memory and created the data in HEXADECIMAL format. One problem which was inherent in all these programs was the danger of memory conflicts, where the original ML would be overwritten by the Basic statements.

DISK, TAPE or MEMORY USAGE

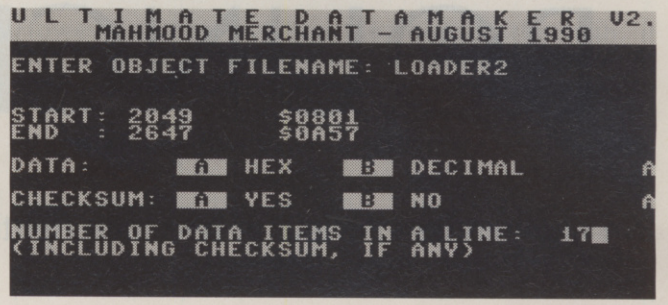
'ULTIMATE DATAMAKER' was written with the above problem in mind. Implemented entirely in Machine Code, it offers the user the option of either generating data from MEMORY or from DISK or TAPE. If the ML is stored at such an address in memory where it will be overwritten by the DATA statements, then it may be saved to DISK or TAPE using any standard ML monitor. 'ULTIMATE DATAMAKER' would then load this file to the unused RAM at 40960 ($A000) under the Basic ROM, thus ensuring that there are no memory conflicts between the ML and its corresponding DATA statements.

TOTAL CONTROL

'ULTIMATE DATAMAKER' allows the user to control such features as the number of data items in a line, whether in 'HEX' or 'DECIMAL', the starting line number and the increment. There is also an option for creating a line-by-line checksum for all lines of DATA plus the facility to append the DATA statements to a Basic program in memory - possibly a loader for the data.

THE UTILITY IN USE

To use 'ULTIMATE DATAMAKER', enter LOAD"ULTIMATE DATAMAKER",8,1 once loaded type NEW followed by SYS49152. Alternatively, select it from the CDU MENU. If you wish to APPEND the DATA statements to your own program, first load it in before activating the utility with the SYS 49152 call. You are asked to enter a filename. The program keeps track of the last device used and attempts to read from that device. (This is for people that wish to keep a back-up copy of the program on TAPE). Hence, if you had loaded the program from TAPE, 'ULTIMATE DATAMAKER' would read the ML from TAPE. Location 186 ($BA), which indicates the last device accessed, is used to decide whether the ML is to be read from TAPE or DISK. If you wish to generate the DATA statements directly from some ML already in memory, then press RETURN at this


ULTIMATE DATAMAKER V2.
MAHMOOD MERCHANT - AUGUST 1990
ENTER OBJECT FILENAME: LOADER2
START: 2049 $0801
END : 2647 $0A57
DATA: A HEX B DECIMAL
CHECKSUM: A YES B NO
NUMBER OF DATA ITEMS IN A LINE: 17
(INCLUDING CHECKSUM, IF ANY)

prompt.You will then be requested to enter the starting and ending addresses of the ML in DECIMAL format. After you press RETURN at the prompts, the corresponding HEXADECIMAL equivalents of the values will be displayed on the screen for your reference. When accessing from TAPE, position the TAPE just before the start of the ML and enter anything (other than only the RETURN key) for the prompt. The filename is not used for TAPE access. Instead, the first program found is read.

FINISHING OFF

Once the ML is read by 'ULTIMATE DATAMAKER', it displays the original start and end addresses. You are then prompted to indicate whether you wish the Data to be in 'HEX' or 'DECIMAL'. The next choice is of a checksum. If this option is chosen than a checksum of the data items in every line is calculated and positioned as the last item in each line. The checksum is calculated by adding the data values in a line and then ANDing the result with 255.

The number of data items in a line is then entered which would usually be 8 or 16 after which the starting line number and increment for the data has to be entered. Please note that if a Basic program is in memory then you will not be allowed to enter a starting line number that clashes with the program. After all these values are keyed in, 'ULTIMATE DATAMAKER' generates the DATA statements in a few seconds. As a speed-test, I used a 30 block ML program, ULTIMATE generated HEX DATA statements for this program in a mere 2 seconds.

HELP AT HAND

To get you started, I've included two sample loaders to be loaded in memory before activating 'ULTIMATE DATAMAKER'. LOADER1 POKEs back data with no checksum whilst LOADER2 does the same for data with checksums. In both programs line 10 defines SA and EA as the start and end addresses which should be changed as according to what is displayed by the 'ULTIMATE DATAMAKER'. In addition, in LOADER2, an additional variable, DT, also has to be specified. This is the number of data items in a line (including checksum). After the DATA statements have been generated and line 10 modified, the full program can be saved to disk or tape using the SAVE command from Basic.

microtyne BOUNCE BACK

An independent review of two men fighting back against the odds - STEW CANEL

One of the major problems that I as Editor of CDU, and you the readers have, because of our love for the serious side of our C64's, is the lack of support from Software Houses and Software Outlets. I receive hundreds of enquiries from you all asking where can you obtain software and peripherals for your machines. If you live in the Northeast, here is good news. (Indeed, even if you don't, I'm sure MICROTYNE could be of some help). STEW CANEL has provided me with a story that is both interesting to read and can prove helpful to us all.....Read on to find out more.

Like a PHOENIX rising from the ashes, **MICROTYNE** of Newcastle opened its doors for business again at 9.30 on Saturday the 20th of October (1990) .

That in itself may not appear at first glance to be a historic event, but it is just the latest move in a small business that has had more than its fair share of ups and downs in its short 2 year life.

Brian Dixon, a former aircraft technician, decided in 1988 that a business selling computers and associated equipment was to be the life for him, so he resigned his secure, pensionable position and opened a small 12' x 12' retail outlet in the Leazes Arcade in Newcastle.

Given his surname, Brian decided that the shop would be known to the world as Dixons Computer Centre, and he duly opened for business. Some very strange things happened during the first week of business, with some very odd looking people taking an interest in his shop, but not actually coming into it, and others taking photographs of it for some reason.

The reason for those photos became patently obvious to him, when a few days later he received a very official looking letter from a firm of solicitors representing the high street electrical giants 'Dixons' accusing him of trading off their name, and instructing him to change the name of his shop.

They also demanded that he sign an undertaking that he would not stock similar lines to their clients, and that he would not use the name 'Dixon' in any form at all, and failure to agree would result in court proceedings.

Being his birthright, this demand was refused, and for the next few months his solicitors and theirs sent protracted letters to each other regarding the point.

Later in 1989, Brian's eldest brother Bill was made redundant from his job as a Compositor at the local newspaper, the job having been all but replaced by modern newsprint technology, namely a computer. So after a modicum of debate, Bill decided to take advantage of his new found unemployed state, and he went into partnership with Brian.

Having had the problem with Dixons earlier, they discussed the possibility of changing the business name, the first suggestion to be put forward, jokingly, was Brian and Bill's Company Computers, or BBC Computers for short, but they thought that this would only perpetuate the problem, and so, on moving to larger premises within the Leazes Arcade, **MICROTYNE** was born on August 1st, 1989.

In March 1990 Dixons withdrew their complaint when they realised that the brothers would not sign their demand over the name Dixon but had in fact adopted the name **MICROTYNE** for their business and everything in the garden was rosy. The business ticked along merrily until the morning of Monday the 16th of July this year (1990) .

It was on that fateful morning, as the two brothers were driving to work that they learnt from the

local news on the radio that the entire shopping arcade had been the subject or an arson attack, and was completely gutted.

Shocked and stunned, they arrived at the arcade to find the fire brigade still damping down the building. When the firemen had completed their task, the shop owners were allowed into the premises to assess the damage and to salvage what little they could.

The **MICROTYNE** shop itself had not been as severely damaged as the rest of the arcade, thanks in the main to a brick wall that part protected it, but the heat, smoke and water damage appeared to have wiped out their entire stock.

At first glance, it seemed as though they may have salvaged some of the boxed computers, as the boxes only seemed to be slightly charred, but when they opened them up, all the plastic casings around the computers had melted and fused together.

Brian and Bill were then able to get a short lease on a small lock up unit to store what was salvaged, and to test things to see if anything still worked.

As if things were not bad enough, the lock up they obtained was broken into once, and had three attempted break-ins in five days, so needless to say, they removed all their stock, and decamped to Bill's garage at his home. They also had their shop telephone number transferred to their homes so that they could at least try and do as much business as possible.

As was expected, the majority of the stock they salvaged was utterly useless, until they set up an Amstrad PC2086 system, and they were more than surprised when the screen flickered into life. They then fiddled with the melted remains of the keyboard, and whatever they typed in, came up on the mangled screen. They then dried out what remained of the printer and coupled it up to the computer, and when they set it off on a print run, they stood back in stunned surprise when it began to print everything they put into it, so never let it be said that Amstrad systems do not stand the rigours of time and everything else that is thrown at them.

They then set about trying to find another suitable premises, and after a good deal of searching they eventually found a much larger and better positioned outlet for their wares.

Originally a bank, and more recently a travel agents, number 1 Gallowgate is ideally situated, just outside Newcastle's Eldon Square shopping complex, and in a very prominent corner position, where almost every bus destined for West Newcastle, and South West Northumberland passes by their front door.

Work to convert the premises into a retail outlet began and the date of the 20th of October was set for the grand re-opening.

However, when I visited the shop on the 19th, it was difficult to imagine them being open a month after that day, let alone the following morning.

A new ceiling had been installed in the shop, new carpets had been laid throughout, a counter had been built, and shelving and display stands bounded the shop, but also, a prolific abundance of computers in boxes, games, software, leads, printers, periphery bits and bobs, and advertising literature seemed to completely cover the floor, and it was hard to imagine any semblance of order ever emerging from what appeared to be complete chaos, but with typical Geordie resilience and their natural persuasive powers, they managed to rope in the help of anyone unfortunate enough to pop their heads around the front door, which is how I came to be still in the shop at 10.30 p.m. that wet and windy Friday night.

Slowly but surely and inch by inch the shop began to take shape, and after alot of the packing boxes had been emptied and disposed of into the back of a van, the shop began to take on an amazing transformation into a sleek looking professional outlet.

With display computers in one section for full demonstrations, games and software set out on shelving in an easy to view way, peripheral items such as leads, connectors, and adaptors etc. displayed on stands, and items such as joysticks, disk boxes, mouse pads, and sheet feeders laid out on shelving, the shop looks decidedly uncluttered and easy on the eye.

Although they particularly specialise in Amstrad and Commodore set ups, they also stock peripheral items, games and software to cover almost the full range and spectrum of computers available.

As well as being in a position to supply almost any kind of new computer or related equipment, the business is also in a position to offer a reasonably comprehensive repair service for everything from the very basic to the very complex of systems.

And so it came, that at 9.30 a.m. on the 20th of October and in nice time for the Christmas demand, the doors opened to the outside world and **MICROTYNE** rose like a PHOENIX from the ashes, and hopefully for the meritorious Dixon brothers it will go from strength to strength.

LABELLER 64

Let your C64 take the hassle out of printing name and address labels. Your C64, printer, address labels and this program are all that you need - A.J.LENTON

How often have you had to type the address of a friend on to an envelope? Or perhaps you have a list of people that you need to send letters to quite often. LABELLER 64 will take the hassle out of this by keeping a record of the names and addresses that you use most often and printing a selection of them out at your command.

LOADING PROGRAM

To load type LOAD "LABELLER 64", N where N=8 for disk and 1 for tape. The enter RUN: to start.

The first screen gives the option of altering the preset width and height of the labels and also the volume of the warning tone.

After these prompts have been answered the main menu will be displayed:

Load Data.....1)
Save Data.....2)
Enter Data.....3)
Search Data.....4)
Alter Data.....5)
Delete Data.....6)
Print Data.....7)
End Program.....8)
Enter choice

LOAD DATA

To load an existing file from tape or disk press '1' followed by 'RETURN'.

You will be asked:

ARE YOU USING TAPE OR DISK? T/D

Press 'T' or 'D' as required.

TAPE

If you are using tape the message:

POSITION TAPE CORRECTLY, THEN RETURN MOTOR WILL STOP AUTOMATICALLY

will be displayed. Next you will be asked to:

ENTER NAME OF FILE TO BE SAVED/LOADED OR 'Q' TO RETURN TO MENU

Enter the file name and press 'RETURN'. The cassette will start and the file will be loaded.

DISK

If using disk you will be asked to enter the drive number and prompted with 0. Amend if necessary. Press 'RETURN'.

You will then be asked:

IS DISK INSERTED Y/N

If you press 'N' you will be returned to the main menu. If you press 'Y' you will be asked:

DO YOU REQUIRE DISK FACILITIES? Y/N

If you press 'Y' you will be passed to the Primary Disk Functions (see below for an explanation of this facility).

If you press 'N' you will be asked to enter the name of the file to be loaded . If the file does not exist an error message will be shown and you will then be returned to the main menu. PLEASE NOTE: Only enter the original file name. Do not add the suffix '.L' as it appears in the disk directory. This is just an aid to identifying 'Labeller 64' files from anything else on the same disk.

CHAINING FILES

It is possible to chain files together. As each one is loaded the total new file will be sorted into order based on the first line of each label. If the memory becomes full the message:

Memory Full
1) to Print Out
2) to Save Data
3) to Delete Data
Enter Choice

will appear. Select the required option.

DISK FACILITIES

If you chose to move to disk facility routine the following menu will appear:

PRIMARY DISK FUNCTIONS
VALIDATE
INITIALISE
NEW
READ DIRECTORY
QUIT
PRESS V,I,N,R OR Q

If you wish to validate a Disk press "V" and the message:

VALIDATING
PLEASE WAIT

will appear.
If you wish to Initialise the disk drive press "I" and the message:

INITIALISING DISK
PLEASE WAIT

will appear.

If you wish to New a disk i.e. format a new one or reformat an existing one press "N". You will be asked:

ENTER DISK NAME

Enter the name that you wish to call the disk. If you pressed "N" by mistake just press return and you will return to the menu for this section. The next prompt is:

ENTER DISK ID NUMBER IF REQUIRED

If you are formatting a new disk enter any two character ID. If you are reformatting an old disk you may just press "RETURN" and the disk will be renamed and the directory cleared. If you press "Q" you will be returned to Edit mode.
If you have chosen to read the Disk Directory it is printed on the screen. To temporarily stop the printout press the "SPACE" bar. To restart the display press the "SPACE" bar again. After the whole of the directory has been printed the following message appears:

RENAME
SCRATCH
OTHER
QUIT
CONTINUE
ENTER:
R,S,O,Q,C

If you wish to rename an ".L" file press "R" and you will get the prompt:

ENTER ORIGINAL NAME

Enter the name of the file that you wish to rename. You will then be asked to:

ENTER NEW NAME

Again enter your choice and you will be told that the program is:

RENAMING "old name"

If you fail to enter either an old name or a new one the program will exit this function.

If you choose to scratch a file you are asked to:

ENTER FILE NAME
When you have done this the chosen file will be scratched from the Disk.If you press "O" you will be returned to the "Primary Disk Functions" menu, if you press "Q" to quit this mode you will be returned to Edit mode and if you enter "C" to continue you pass to the part of the Disk program which asks for you to enter the file name.

SAVE DATA

To save a file to tape or disk press "2" followed by "RETURN".

The prompts are then similar as for loading. When positioning the tape press "RECORD & PLAY" before pressing "RETURN". When saving to disk you are given the option:

DO YOU WANT TO CREATE A NEW FILE OR OVER WRITE AN OLD ONE? (N/O)

If you press "N" and a file with the same name exists you will get an error message and you will be returned to the main menu.

ENTER TEXT

To enter text press "3" followed by "RETURN". The screen will show the outline of the label, the number of the current label being entered, the number of labels that the memory will accommodate (variable depending on the size of the labels), the number of lines on the label and the width of the label. To abort entry of text and return to the main menu press "F7". This may be done at any time during the entry of text.

Entry of text is as normal and editing and cursor keys work as normal with the following exceptions:
To insert text place the cursor at the position where the additional text is to be inserted. Press "INST" (SHIFT/DEL) and a reverse "I" will appear, in purple, at the bottom right corner of the screen. Any further text now entered will be inserted between the text to the left of the cursor and the text starting under the cursor. To turn this mode off press "INST" again.
To move the cursor swiftly to each end of the line being entered press "CTRL" and "<LEFT ARROW>" together. If the cursor is not at the start of the line it will be moved there. If it is at the start it will be moved to the end.
At the end of each line press "RETURN" and the cursor will move down to the start of the next line.
The "UP" and "DOWN" cursor keys will only work if there are lines either above or below the one on which the cursor is present.
To complete entry of text either press "RETURN" continuously or press "F1" (the latter can be quicker if the label has been set for a large number of lines and you only wish to enter text on the first few).

In either case the message:

Is This Correct (Y/N)?

will appear. If you wish to modify the text press "N" and the cursor will reappear.

If you press "Y" the message:

Another One (Y/)?

will appear. If you press "Y" the screen will clear ready for the next label.

If you press "N" you will be returned to the main menu.

A tone will sound at five characters from the end of a line and also at the end of the line. Also if you try to enter too much text on to a line a warning will appear:

LINE TOO LONG
Please Alter
To set auto repeat on all keys press "F3" and to cancel press "F5".

SEARCH TEXT

To search for a label press "4" followed by "RETURN". The screen will clear and you will be asked to enter the search details. During this entry only the cursor left and cursor right keys can be used to move the cursor through the text. However, insert and delete work in a similar manner to the main entry of text.

When you have entered the details press "RETURN". The

screen will clear and the message:

SEARCHING

will appear at the top of the screen. The program will then search through all the labels in the file and will list the number and first line of all labels which include the search details. Should the list reach the bottom of the screen the listing may be stopped by holding down any key. Obviously the more detail entered in the search pattern the fewer matches will be found.

When all valid entries have been found you will be asked to enter the number of the label to be viewed. Enter the number and press "RETURN". The requested label will be displayed and you will be asked if you want to see another one. If no valid entries are found the message:

ENTRY NOT FOUND
Do you wish to try again (Y/N)?

will be displayed.

AMEND LABEL

To amend a label press "5" followed by :RETURN". Proceed as for SEARCH and when the required label is displayed it may be amended as per entry of data.

DELETE A LABEL

To delete a label press "6" followed by "RETURN".

Proceed as for SEARCH until the label is displayed. The message:

IS THIS THE ONE TO BE DELETED Y/N

will appear. If you answer "N" the "Enter Search Details" screen will reappear.

If you enter "Y" the label will deleted from the file and you will have the opportunity to delete another one.

PRINTOUT LABELS

To printout labels press "7" followed by "RETURN".
The screen will clear and you will be given the option of either choosing the labels to be printed (useful if you only wish to print a small number from the file), choosing the labels NOT to be printed (i.e. if you wish to print most but not all the labels), or printing the whole file:

Do you wish to select
1) ADDRESSES TO BE PRINTED
2) ADDRESSES TO BE OMITTED
3) TOTAL PRINTOUT
Enter Selection Required

If you choose either "1" or "2" the screen will fill with the number and first line of all the labels in the file. Enter the appropriate numbers and press "RETURN" one at a time. If there are more than 20 labels in the file you can enter "C" to continue the listing. If you do not wish to see anymore press "P" to go to the printout routine. If you choose "3" you are passed straight to the printout routine.
In the printout routine you are first given the choice of printing to the screen or printer. If you select screen, a screen full of labels will be printed.
If you select printer you will first be given the opportunity to alter the device number of the printer (it is preset to 4). This is followed by the secondary address of the printer (preset to seven for lower case mode).
Next you are asked:

Do you want
Manual or
Auto paper feed
Enter "M" or "A"

The next prompt is:

Number of lines to be printed 11

where "11" is the total number of lines set for the label. If you wish to print less lines, alter the number and press "RETURN".

You are then asked if you are using single or double width labels. Press "S" or "D" as appropriate.

Finally you are requested to enter the width of the paper in inches and prompted with "g". Amend this if necessary (this only applies if you are using double width labels and the measurement is taken from the extreme edges of the paper including the tractor perforations).

If you wish to interrupt the printing at any time hold down the "SPACE" bar until the message:

Do You Wish To Reposition Paper or Stop Printing

appears on the screen. To stop printing press "S" and to reposition the paper press :R:.

EXIT PROGRAM

To exit the program or clear the memory for further entries press "8" followed by "RETURN".

You will first be asked:

ARE YOU SURE (Y/N)?

If you press "N" you will be returned to the main menu with the existing file intact.

If you press "Y" you will be asked:

Do you want to run again (Y/N)?

If you press "Y" the memory will be cleared and you will be returned to the initial screen.

If you press "N" the program will erase itself.

LATE AMENDMENTS

Since the copy for this article was set the author has made some alterations to the program. These are as follows:

1) The files are no longer sorted into order. This saves time during the operation of the program and also some memory.

2) It is now possible to exit a LOAD routine by holding down the SPACE bar.

3) If using disk it is also possible to abort a SAVE in the same manner and any file created will be scratched.

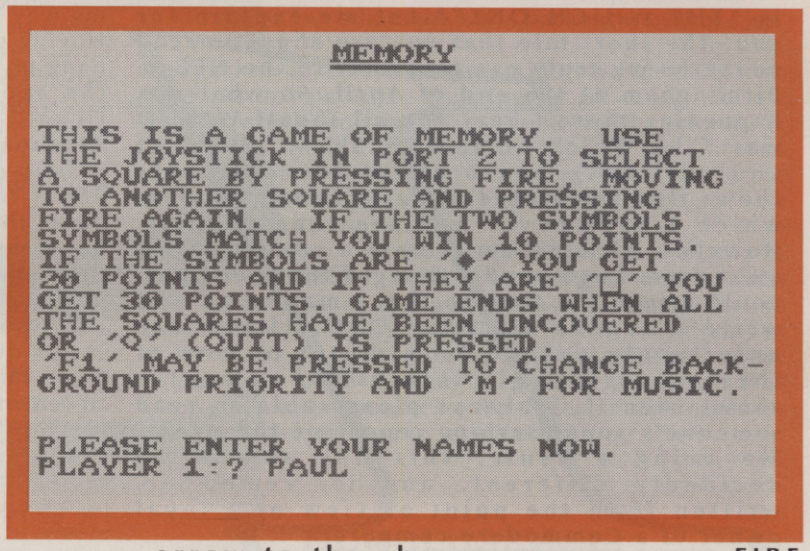
4) When the program is in search mode it is now possible to stop the search by press 's'.

5) When printing out labels the program now asks for the number of characters to be left between adjacent labels and it is now possible to set the number of lines between labels.

MEMORY

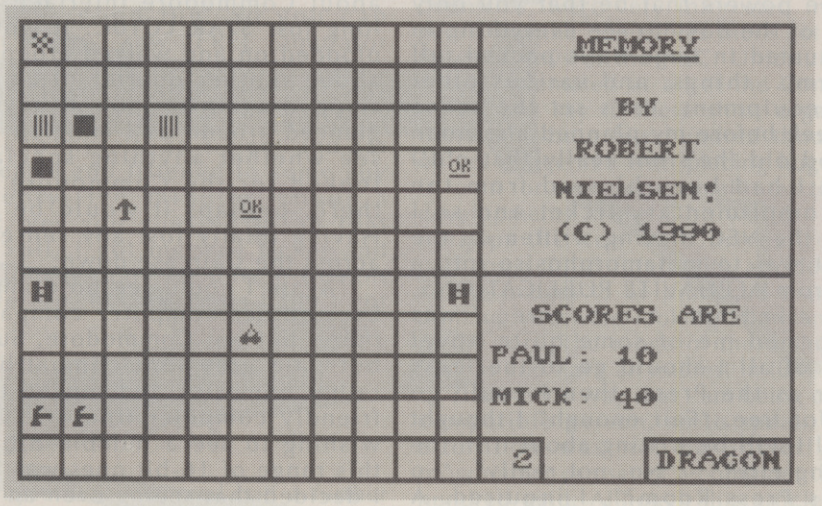
A simple game of concentration for those fed up with mayhem - R.NIELSEN

We have all played concentration with a pack of cards at some time or other, and so I thought I would design my own computer version to while away the hours when it's raining. The game is played on a board with 120 squares. Each square is occupied by a symbol (which is obviously covered). Players take turns in trying to pick (uncover) two matching symbols.


MEMORY
THIS IS A GAME OF MEMORY. USE THE JOYSTICK IN PORT 2 TO SELECT A SQUARE BY PRESSING FIRE, MOVING TO ANOTHER SQUARE AND PRESSING FIRE AGAIN. IF THE TWO SYMBOLS SYMBOLS MATCH YOU WIN 10 POINTS. IF THE SYMBOLS ARE '♦' YOU GET 20 POINTS AND IF THEY ARE '□' YOU GET 30 POINTS. GAME ENDS WHEN ALL THE SQUARES HAVE BEEN UNCOVERED OR 'Q' (QUIT) IS PRESSED. 'F1' MAY BE PRESSED TO CHANGE BACK-GROUND PRIORITY AND 'M' FOR MUSIC.
PLEASE ENTER YOUR NAMES NOW.
PLAYER 1:? PAUL

PLAYING THE GAME

Load the game with LOAD"MEMORY",8,1 and RUN (Or select it from the CDU menu). The MUSIC and CHARACTERS will then be loaded in and the introduction screen will appear. Type in the names of the 2 players and the Main Game Screen will come on. Wait for the counter at the top left of the screen to get to zero before starting. To quit at any time press 'Q' and if you do not like the music, press 'M' to turn it off. 'F1' toggles if the ARROW is above or below the GAME SCREEN.



Plug your joystick into Port 2 and start playing. To pick a square, move the arrow to the chosen square, press FIRE and the symbol will appear in ORANGE. (The ARROW will turn GREY) Now move the arrow to the second square and once again press FIRE. If they match, you will get points. 30 for a non-filled in square, 20 for a diamond and 10 for everything else. If a player gets a match the game stays with them until they do not get a match.

If you forget whose turn it is, look at the little box under the scores. A 1 means it's player ones turn and a 2, player two. The game ends when either 'Q' is pressed or all the squares are uncovered. My best score to date is 1220!! Good luck.

WHICH ONE

One mans opinion on the recent NEC show

What then is this article all about? Which one is "THE WHICH ONE"? Furthermore, what is "THE WHICH ONE"? Let me explain for you. The short tale that you are about to read concerns my truly exciting visit to the NEC in Birmingham at the end of April. So what was happening there I hear you all shout! Well, it was "The Which Computer? Show". Many of you will have heard about the computer shows that are staged at Earls Court, London, which I have never had the time to visit. However, Birmingham is only a stone's throw away from Rugby and I thought that perhaps I could afford the time so that I may be able to write a report of it for CDU. If you are unfortunate enough to miss out on one of these devastatingly interesting computer shows then it is always pleasurable to read someone's super-serious report of the event. Me being a cynic, this one should be decidedly different, and is, remember, written from the point of view of a loyal owner of a Commodore computer.

THE SCENE IS SET

These shows usually last about four or five days (although thank the powers that be that you only have to go on one of the days to see everything in action) and are situated in huge halls packed full of electronic... erm... things, and various other computer related equipment. Let's set the scene starting about a week before my planned afternoon out. I told the good old (he's not really that old!) Editor, Paul, that I had been blessed from the heavens with a complimentary ticket and so I would be going to the NEC, having written on the piece of card, that was to metamorphosise into a badge, that I was from ALPHAVITE PUBLICATIONS. "Got any tips?", I asked him. "Yeah, sure..." was his reply. He informed me of some quite novel things to do. First of all I should go to the Press Room and register as the Press, then I could go back another day for free. "Fair enough" I thought to myself - after all I will be writing about it in the magazine. "Anything else?" - "No, not really". "So how do I find this Press Room?", I inquired. A quite obvious reply was given - "Just ask the person who takes your ticket - it's usually upstairs somewhere. It is at Earls Court isn't it?" - "No Paul, I said the NEC" - "Oh, well they all have them so just ask like I said". "Great" was my first reaction. The thought of getting something for nothing made my eyes light up and when the day came I jumped on the first train I could and made my way to Brum.

Having paid my fifty pence for a group of ladies to hang my jacket on a coathanger, I made my way to the appropriate hall entrance and gave in my ticket. It was at this point that my piece of cardboard was miraculously transformed into a plastic-coated badge. So I asked the young lady about the Press Room. "Oh, I'm not quite sure about that, Sir - you'll have to ask at Information". So then I had to find the information desk. Having done that they also didn't have a clue what I was going on about. It seems that I have found an exception to the rule of Press Rooms - or perhaps it was outside somewhere. Never mind, by the time that I had left later on I was glad that the incentive to return wasn't there anymore.

WHAT NO COMMODORES?

So what was there at this show? I wandered around and bumped into the Star Micronics area. This was actually interesting (and now I'm not being sarcastic) and I picked up a booklet all about Star printers and their prices. This I thought would be useful. It's bound to have information about Commodore interface printers such as the long lost STAR LC10C. It would appear not. So I plodded on and walked around the upper level for about another quarter of an hour. I never really stopped again to look at anything in particular. I glanced around as I mingled in with the crowd to see whether anything special caught my eye. I looked for the Commodore stand but it wasn't there, perhaps it would be on the lower level. Having totally lost any sense of direction I came across the staircase down.

Again I could see no Commodore stand. What? Every other computer company in existence was there, but not Commodore. Surrounded by nothing but pale grey IBM compatibles I was becoming rather homesick and longed to see a decent user-friendly computer from the Commodore range. If wishing to see a humble 8-bit computer amongst this range of 16-bit ones was hoping for too much, I decided that seeing even an Amiga 500 would be nice, or maybe an Amiga 3000! No such luck. I had read in a company's catalogue that at a computer show in the past everyone had marvelled at a high resolution picture image and how a C128 computer could produce it. There was nothing so spectacular this year. On my travels, I came across

many different stands that were plugging their latest products, all IBM compatible gear for the super-serious office team.

A CLEAN MACHINE

One thing that I actually stopped to look at on the lower level was a large tank of bubbling water. But there was something rather strange about this tank - inside it was a computer, together with a disk drive and a monitor! They were all stripped down so that through the murky depths you could see the electronics, and what's more, all three pieces of hardware working - underwater!! There were no optical illusions anywhere - this set-up was truly operating surrounded by that common fluid. I had to laugh at the little warning sign - "Caution: Do not try this with your computer system". This it seems was to be my penultimate dose of humour that afternoon. Please don't write to Techno-Info asking how to get your C64 to operate in the bath because I just don't know how it is done.

TO SIM OR NOT TO SIM

Having wandered around aimlessly for a bit longer I saw one of those large white machines, commonly termed a "simulator". When I had been traversing around Covent Garden in London a couple of weeks beforehand (before going to see "Miss Saigon" in the evening - see, I'm cultured!) I had seen something similar and decided to waste two fifty for a two minute ride. You sit in it with a group of other people, crammed four to a seat that looks as if it were designed for just two. A film plays at the front and the hydraulic pistons underneath shunt you around to mimic the movements that would be going on if you were really on the rollercoaster, or in the RAF aircraft, and so on. It really is good fun. So you can understand my relief when I saw "FREE Simulator Ride" plastered on a big poster nearby. Then I read the small print - "Just attend one of our presentations". What a small price to pay for a good laugh I thought to myself. So I went and sat down in the little theatre and listened to some guy babble on for twenty-five minutes about a database system for, yes you guessed it, an IBM compatible system. Still, I got my free ticket and got a ride. It lasted longer than the one in Covent Garden and was more fun, but the seats hadn't got any bigger!

STALLS FOR ALL

Anyway, I pulled myself together and had a further look around. In case you have never been to one of these electronic extravaganzas, I shall tell you that other than the leading firms, many small companies have equally small "stalls" where they sell their goods and advertise themselves. I managed to pick up a great bargain of 1000 self-adhesive tractor-feed address labels. I had been contemplating a purchase of this magnitude for yonks and decided that a fiver was a fair enough price to pay. I then stopped off at a stall marked "Hong Kong 2". They were selling 3.5" and 5.25" disks made up as clocks! Whatever next - though the temptation to buy one was definitely there. You could buy them there or send off for them to their company in Hong Kong. After much deliberation I decided on the third option and continued to walk past. By this time my mind was going dotty and I was curious about such silly things as where "Hong Kong 1" was. Anyway, I found it and they were selling computerised personal organisers.

Now, I had gone up and down almost every aisle and yet I kept coming across new things that I hadn't seen before. I was about to break down in utter despair but then I found a notice board with one of those familiar "You are here" arrows stuck to it. A map of the hall! I tried to work out where the stairs back up were but I just couldn't find them. Every door on the lower level that was marked as "EXIT" had a security guard blocking it so I thought that perhaps I would never get out. I had lost track of which directions I was going and I couldn't see above the huge cardboard facades of the various companies, but eventually I found the staircase. On the upper level it was just a matter of finding the proper way out back into the rail station.

Armed with my thousand sticky labels, a booklet and a bunch of leaflets that had been thrust into my face from mystery hands as I walked past, I searched for my closet ticket. Eeek! I thought I had lost it - I had visions of having to go back into that hall and searching every square inch until I found it. Luckily the ticket was screwed up in my trouser pocket and so on this occasion fate must have decided to let me off!

CONCLUSIONS!!

Now don't get me wrong here - I'm not trying to tell you never to go to one of these computer shows, because I feel that they can be very valuable, so long as you've got an IBM compatible interest. Many of the people that were walking around, and they were from many different countries, were very interested in the products and I guess that the companies there got a lot of sales out of the four days at the NEC. Unfortunately this show just wasn't orientated at all at the Commodore audience. I would just say to these show organisers that although there may be a few hundred thousand IBM users and offices concerned with placing huge orders, surely there is room for even just a small area to be set aside to new developments in the 8-bit world. I had expected to see perhaps the new hard drives from Commodore, or perhaps a C128 fitted with a Video RAM Upgrade that allows images to be produced that could fool you into thinking you were looking at an Amiga display. Nothing of the sort. The range of these shows is vast, again so long as you have an interest in IBM compatibles.

There is no way that I am being converted into an IBM compatible person, so to speak, and I feel that this afternoon was not particularly worthwhile to a programmer or user of a computer that cost less than five hundred quid. Although this article may seem a bit one sided, there is objective argument concealed within the text. A talented reader should be able to decide which side of the dividing line he falls when considering going to the next big computer show that rears its ugly head. Just ask yourself whether or not you have, on your person, a concealed 24 pin D-plug that makes you fully IBM compatible.

CDU

GAMES SPECIALS! SOFTWARE OFFER

Fed up of paying huge amounts of dosh for your games??? Let CDU remedy this by offering you these superb games compilations at knock down prices

All of the disks on offer are original, never before seen games. There is something for everybody. (Shoot 'Em Ups, Strategy, Adventure, Mind Benders and straight-forward Platform). No matter what your preference, something, somewhere will take your fancy. To order your choice, simply fill in the coupon below and send it with your Cheque/Postal Order (made out to ALPHAVITE PUBLICATIONS LTD) to:- Alphavite Publications Ltd, 20 Potters Lane, Kiln Farm, Milton Keynes, MK11 3HF. (Please allow 28 days for delivery).

GAMES DISK 1 (1991)

CONFUSION - So you think you are quick witted? Think you are of high IQ? Crosswords don't hold enough interest for you because they are lame ducks for your mind? If you answered yes, or even no, to those questions then Confusion is for you. A two dimensional version of the popular cubic puzzle. Sort out the multicolored columns - simple? Ha, try it.

TENOGEN - Blast almost everything in sight. By destroying whole waveforms you will increase the amount of extra weaponry to collect later in the level. Eight scrolling levels to destroy takes you to the end of this exciting shoot-em-up, but can you reach the end?

PROJECT X - You play the part of Hank, in this graphic adventure. Hank sole purpose in life is to retrieve the secret documents of project x. There are four very tricky stages to get through in your quest. First you must fly your plane, then land it after which you must run along the beach and climb a cliff, through the jungle, until you find the cave where enemy agents hold the document. Avoiding enemy aircraft, falling boulders, spears, and arrows - phew, can you find the hidden message...?

MEGADOGFIGHT - An aeriel combat game for two players. Guide your plane around the screen and try to shoot down your best friend as he pilots his aircraft around the screen trying to shoot down you... Great game for two people out for a Sunday flyabout.

GAMES DISK 2 (1991)

FAST FUTURE - This is an arcade type game where you take control of your craft and guide it around a circuit a set number of times - oh, if life was as easy as that. Indeed not, there are other craft in the 'race' who plan to give you more than a really hard time. However, being a bit of a b..... yerself, you blast 'em with your twin lasers, as well as bumping them outa existence. Banks, gravity tracks, collecting energy shields, 32 levels, and ....

COLD COMFORT - In this adventure you awake to find yourself alone on an alien space ship, and locked inside a holding cell. Your task, should you accept it, is to escape the cell, learn the alien language, and discover how to pilot the 'ship' back to earth. This text and graphic adventure will keep you pleasantly engrossed for hours. By the way, it is a big ship.

CELLRATOR 11 - The sequel... as you can guess this has the same theme as cellrator but try and beat this one. Scrolling screens of caverns and caves and never ending obstacles as you fly your craft along; heavy foot on the accelerator, getting you into all sorts of collision trouble, making you wonder if it is all worth it. Quite frantically yes it is! Make map??? Ho! Ho! Ho!

ERADICATOR - A very colourful, with beautifully designed graphics, screen scrolling arcade type game. Survival is the name of the game as you try to avoid all contact with other lifeforms - and just what good are your lasers, I'd like to know? Anyway, can you save the earth, yet again! By the way, slimy green aliens are running the world governments and only you know this, but who would believe you anyway - that's why you grabbed your battlecruiser in the first place!

GAMES DISK 3 (1991)

SOLSTICE - This is a three part graphic adventure set deep within the fourth and largest moon of some distant planet. This game will tax your brain with its complexity as you try to reach completion in the third and final part. You will have to kick, punch, dive, roll, and run your way through each screen, all the while keeping your eyes open for clues. Remember, the diamond must be destroyed!

NEW YORK CRISIS - New York has a problem... The computer of NY surface defence missile silo #5 has declared war on the city. As you are Controller, on of the elite trouble shooters in the city, you must assemble a team of three to enter the silo and disable it. No easy task. If you like games of strategy where fast thinking is of utmost importance then this will leave you with weeks, maybe months, of enjoyment.

GAMES DISK 4 (1991)

LIFE - There have been many 'Life' programs created for the computer since John Conway toyed with the idea of a mathematical model of the behavior of living cells in the 1950s. Here is another version, but this time for the C64, and within

which you have the ability to bring to 'life' dead cells. An interesting variation of the theme of life.

WHITEWASH - This is a logic game where the objective is to reduce the counters to white by successive hits before your opponent does the same. The game is based around the C64's ability to show colour on the screen, and the idea is basically to strip off various layers of colour until white is found.

FRUSTRATION - Is a variant of the old hand-held moving tile game. The aim of the game is to arrange all of the tiles in such a way so that they form the picture shown on the right hand side of the screen.

EUCHRE C128 - This C128 game, which works in 80 column mode, is based on the old card game of the same name. You play with a computer partner against two computer opponents.

HYPERSOLVE - Erno Rubik's cube finds its four dimensional equivalent on the C64. Yes, you must solve the problem of the hypercube which is a four dimensional object that consists of 16 corners, 32 edges and 24 faces, making up 8 cubes, each of which is adjacent to 6 of the others - phew! Can you solve this one?

BINGO 128 - Yes, Bingo for the Commodore 128. This rather interesting version of bingo will allow you to print your own bingo cards, and then will produce the bingo numbers either manually, or automatically - what this means is that Manually the time interval between the calling of numbers is controlled by the caller and in Automatic mode you are able to preset the time between each call. This is a must for those family and friends get-togethers.

GAMES DISK 5 (1991)

ORB - Ever heard of living space coral? No? Well let me tell you this is pretty deadly stuff and not for the fainthearted to deal with. However, you are not fainthearted are you, so off to battle with the deadly ORBSTAR, but watch out for the nasty aliens who materialise in the most unpredictable of places - still, with your powerball at the ready, you're sure to be a winner - eventually?

LANCE - The island of Brittania has been plunged into the dark ages. The evil witch Morgana has stolen the holy grail. Many brave knights have tried to recover it, now it is your turn.

PROBE WARRIOR - Life in deep space is never running smooth. Just when you think all is peaceable and nice, you have to set forth and defend your planet against the dreaded Clax. You must stop him from destroying the lifepod system otherwise all life on the planet will be exterminated.

LIBERATOR - An exciting all action game with ultra-smooth screen scrolling, and where you, as the liberator, and after being sent to Venus, must liberate the people by clearing the lands of all the invading aliens. You can contact the resistance forces, collect credits to gain weapons such as 'smart bombs', and regain your depleting energy from the rejuvinator tree.

GAMES DISK 6 (1991)

OUTBREAK - This is breakout but with a major difference - the screen scrolls. You must break through the massive play area until you reach the ALMIGHTY wall at the end... On your journey you will meet with aliens, which can be destroyed, life giving blocks, as well as boring, tough, exploding, happy, angry, and deflecting blocks. You will like this one.

THE MYSTERY MAN - Here is a rather snatzee adventure game where you play the down-at-heel private dick with landlord problems and no booze and no customers. Suddenly, into your life comes a man who offers you five-hundred smakeroos just to deliver a cassette recorder to some guy in a downtown hotel. Grabbing the recorder and your gun you head off into the adventure of your life!

MIRROR IMAGE - Message commences. Dateline 2237. Draconian Empire ships heading through hyperspace towards solar system. Danger scale 100% Multiphase reality track now in operation. Mirror image ERU awaiting pilot. Mission, destroy all Draconian Ships which materialises. Message ends. And of course, you know who the pilot is, don't you?

LIBERTE - Here you are, sitting in your hut in the POW camp. You've been there for far too long. A hundred times you have gone over your plan, surely nothing can go wrong. The time as come for you to put your plans into action and escape. It wont be easy though, for a start there are the patrols to avoid, then there is the small matter of the Gestapo HQ to blow up not to mention the rendezvous with the ships Captain. Believe me, I don't envy you in your task.

Please send me.....Copies Disk No. 1 @ £5.95 each .....Copies Disk No. 2 @ £5.95 each
.....Copies Disk No. 3 @ £5.95 each .....Copies Disk No. 4 @ £5.95 each
.....Copies Disk No. 5 @ £5.95 each .....Copies Disk No. 6 @ £5.95 each

I enclose a cheque/postal order for.....
(Made payable to **ALPHAVITE PUBLICATIONS LTD.**)

NAME.....
ADDRESS.....
.....
.....Postcode.....

Send to; **ALPHAVITE PUBLICATIONS Ltd;** CDU Software Offer, 20 Potters Lane, Kiln Farm, Milton Keynes, MK11 3HF.

TECHNO-INFO

Our ever popular technical bit gets more and more mail, here's a few more people sorted out - JASON FINCH

First of all, I must apologise for there being no TECHNO-INFO in last months magazine, I think by now you all know the reasons why. Secondly, I must also apologise to the following people for having to yet again hold their replies until next month;

Craig Dickson, Solihull.
C.D.Roberts, Cornwall.
Stewart Hall, Exeter.
V.Perry, Barry.
N.K.Taylor, Bournemouth.
Maurice Le-Vallois, Paisley.
Alfred Fox, Liverpool.
Richard Viatonu, London.
W.J.Wilson, Sutherland.
Ann Pickston, Manchester.
R.Dunley, Chester.

On with the show!!

So you've looked through the mag and decided to stop here and have a read - good on you. What's happening this month in TECHNO-INFO, I hear you ask. Well, for a start, the usual UPDATE section is here giving you information on past enquiries, together with TIP OF THE MONTH to provide you with yet another invaluable bit of information. And of course there are the letters - this month we HAD sixteen of them just for you, ranging from a query about a BASIC routine to one about C64 Emulation on an Amiga, but as already mentioned, we have had to cut this down dramatically. Please remember that TECHNO-INFO is also a section in which your opinions on certain matters can be expressed - we don't only publish letters from people with computer-related problems. Anyway, I won't use up further space, I'll just let you get on with reading the offerings.

VOTE GEOS

Dear CDU,
I keep buying the magazine out of loyalty really as you're the only magazine still with us dedicated C64 and C128 users, but I hardly use any other program apart from GEOS. I find it does all I want from my computer, almost - which brings me to one of the reasons I am writing to you. I read a letter in the April 1991 edition where one guy says GEOS is far too involved - I just couldn't disagree more. I have Mini Office and though I think it's a very good package, I would never use it again for writing letters. With GEOS I don't put control codes in to alter things - I just highlight and then change it to whatever I want, and Mini Office does not give me a choice of whatever fonts I want or 7 different styles like GEOS. And I certainly cannot add graphics like the ones in this letter. I will agree to make a letter like this takes time but if I had not messed about putting graphics in it, it would have been quicker than other word-processors I have used. All I can say is *** LONG LIVE THE C64 AND LONG LIVE GEOS ***. Now my second reason for writing. I wonder if there is any chance that you could put some of the Public Domain GEOS software on the disks as I am sure fellow users would appreciate that and with your contacts I am sure you would be able to get hold of many of the programs put out on Q-LINK in the States - not only GEOS programs are available so it may be possible to fill the disks up with all sorts of goodies. Finally if there are any GEOS fans out there that want to get in touch to exchange ideas or GEOS utilities, please feel free to write. I endeavour to answer all letters (join the club! - TechEd). Very best wishes to you all.

Frank Cassidy,
55 High Bank Road, Droylsden,
Manchester, M35 6FS.

Dear Frank,
Many thanks for your letter about GEOS. Everyone has their own views on things and if you prefer the environment set up by GEOS then that is fine. Personally, I use SuperScript for my word-processing needs, but that is only because my copy of GEOS-128 has got a most infuriating bug in it which causes spurious little dots and lines to appear randomly on the screen whenever you do something. When using GeoPaint this is particularly annoying because it messes up any pictures, not that I would ever attempt to draw anything. GEOS is a very good package, but so is Mini Office. It all depends on what your needs are, I suppose. Regarding Public Domain software on the CDU disks, what a great idea. We would certainly publish some if somebody was to send it. There

are no hassles about putting PD stuff on the disks, it is just the fact that it is never sent for consideration. It would certainly reduce the production costs. Once again, we appreciate your views and I hope you get lots of people writing to you.

SPEEDIER SORTER

Dear CDU,
I should be most grateful if you could help me on two points. I have written (in BASIC) for my C64 a program for storing, sorting and printing details of music on disks and tapes. I find that, when sorting, the program locks up for periods of several minutes at a time although, if patient, the program will eventually continue and complete the sort. I assume that this is due to "garbage collection" but it does make the sort time very long. Is there any way to avoid this? I did try "Supersort 64" in the October 1990 CDU but I couldn't get it to work. Secondly, is there any way of chaining together two sequential files to form a new file?
Malcolm Mort, Swansea.

Dear Malcolm,
The problem is not entirely due to garbage collection, which is the destroying of all unneeded variable and string information for those that don't know. It can be caused because of the sort algorithm that you are using - perhaps it is not particularly efficient. But the main cause will just be the speed of good old Commodore BASIC and it cannot be rectified too much. The only thing that I can suggest is for you to sort the titles or whatever as they are entered. In that way you will only have to, using a FOR...NEXT loop, move a block of titles down one space to open up a gap in which the new one can be slotted. If a title is amended, the block below it can be moved up one space and then the other routine called to insert the new title in the right place. In that way you have an index that is always in alphabetical order and there is no need to have a sort routine as it were - and it will take a fraction of the time. On your second point, yes there is a way of chaining two SEQ files together, and it can be used to link up to four of them. You must issue the following commands: OPEN 15,8,15,"I0": PRINT#15, "C0:LINKED=ONE,TWO": CLOSE 15. 'LINKED' is the filename for the new combined file, and 'ONE' and 'TWO' are the names of the two **smaller files.**

COLOUR PRINTING

Dear CDU,
I read in one of your back issues (November 1990) that Mr.David Paddison had a problem with "dumping" colour pictures to his printer. We also had a few problems with our system. My dad, after hours of work, finally found that DIP Switch No.5 on the STAR LC10C colour printer had to be down for it to print in colour (Yes, we do have the LC10C colour printer if you are wondering). So, for all you people out there with the STAR LC10C colour printer I hope your problems are solved. Also, thanks to the Techno-Info Squad for writing about the SUPER SNAPSHOT V5 cartridge. Dad bought it for the colour printer option, but I like the games master on it! Hoping this will get printed.
Simon Knight, age 11 (and 3 quarters!!), Northag, BFPO 40.

Dear Simon,
The problem stated by Mr Paddison was slightly different - he didn't have the same printer as you. He had the standard STAR LC10 which does not have a serial port in the side, and so an interface is necessary to enable connection between the printer and computer. The standard model, as opposed the Commodore standard model with is suffixed with a 'C', also has slightly different commands and so totally compatibility between the two is not always guaranteed. But thank you for the information. Some problems can also be relinquished by pulling the release lever at the back right forwards. I am glad that someone found my recommendation about the cartridge useful, but I would hardly call one man a squad! Have fun and I guess by now you will have reached that milestone of ages, twelve. Thanks very much for your letter.

THE 1551 CONNECTION

Could you please advise me if it is possible to convert or adapt the Commodore 1551 disk drive to operate successfully with the Commodore 64 computer. I am aware that this disk drive was originally introduced for the Commodore Plus4 but as my Plus4 is beyond repair, I would like to use this disk drive with the 64 if possible. I hope that you can assist me on this matter.
J.Bond, Rossendale.

Dear Mr.Bond,
I am reliably informed by experience that there is no way whatsoever of connecting the said drive to the 64 and having it working correctly. The 1551 drive will only "talk" to the Plus4 unfortunately. I'm afraid you will have to buy a 1541 or something similar.

VIEWS ON THE MAG

Dear CDU,
I would like to thank everyone involved with CDU for making it such an excellent and informative magazine. Although I don't necessarily always find everything of use to me, there is usually something in the pages that keeps me occupied. I could sing praises all day but I feel that a bit of constructive criticism would be more beneficial. Whereas the "comic" mags for the 64 are too flippant, I find that CDU often goes near to the other extreme and is very serious - not too serious you understand. I would like CDU even more than I do at the moment if a few pages each issue were devoted to something unusual and off the beaten track entirely. I refer to things like "Tomorrow's Tomorrow" and that wacky, very funny story "Noddy's Revenge" that appeared a while back (who actually did write that by the way?). I like series about programming like "Basics of Basic" and "Machine Language Techniques" although having had my computer for six years and delved

into BASIC quite a lot the former is not of much use to me personally, but do they need to be so long? Couldn't series be broken up into smaller chunks so that they occupy perhaps only three or four pages each issue. This would leave room for other features. Personally I would prefer lots of short articles, rather than a few long ones. I remember when Techno-Info was only two pages long - although I'm not saying that that should in any way be shortened because I find it probably the most informative and helpful section of the magazine. And what has happened to all the games? I am sure that everyone would agree that one game on each disk wouldn't go amiss. Those are my views on things - notice that I didn't comment on the price increase to £3.25. This is because I feel that CDU is well worth the money that we pay for it. Keep on with the great work and I wish everyone involved much success with the continuation of a great magazine.
Christopher Dicey, Manchester.

Dear Christopher,
Comments about the magazine are always gratefully accepted. I am glad that you could sing our praises all day, and incidentally it was actually me (yes I confess) that wrote "Noddy's Revenge". I'll let you into a little secret - there may be a sequel to it coming out shortly. Regarding the length of articles, it is not always possible to split them up into small chunks if they are referring to just one topic. The reader may need to get into the 'flow' of the article and not be able to leave off a specific topic for a month, but I would actually tend to agree with you. I try to keep Adventure Writing and Helpline to a minimum to leave space for other things and perhaps more shorter articles would help. You say that you are sure that everyone would agree that one game per disk would be acceptable. I hate to disagree with you but if too many games are published in short periods, we get letters from people asking whether games are really needed. There are two sides to every story - some people would like the odd shoot-em-up interspersed now and then, and some wouldn't. Personally I think that the odd game helps once in a while. I hope that you continue to enjoy the magazine.

UPDATE

Only a short update this month. I would just like to thank everybody that wrote offering cartridges for PETER APPLEBY OF SALISBURY and VIC20 RamPacks for RAY ROBINSON OF DARLINGTON. There are too many of you to mention all your names unfortunately. Many thanks for an overwhelming response.

TIP OF THE MONTH

There are two tips for you this month, the first from MR J.S.MARDELL OF STOKE NEWINGTON:

If after a program in BASIC is loaded, you write down the values given by PEEK(2049), PEEK(2050), PEEK(45) and PEEK(46), then if NEW or a reset cartridge is used and the program is wanted again, you can enter the values back with POKE 2049,x: POKE 2050,x: POKE 45,x: POKE 46,x. Then you can LIST your program again. I have tested programs by placing the cursor on any one of the line numbers of the listing and pressing RETURN. The program still stayed on the screen.

Thanks very much for that Mr.Mardell. There are plenty of people that will find that useful. Now the second one, from **PETER WEIGHILL OF BOURNE:**

1. X=-(X+1)*NOT(X=N) counts from 0 to N, resetting to 0 after X=N

2. X=-NOT-X*NOT(X=N) counts from 1 to N, resetting to 1 after X=N

3. X=-(X=-X)*N alternates between 0 and N (where for 1,2 and 3, N is a positive integer)

4. A$=CHR$(A+48-(A>49)*7) changes dec A to hex A$ (where 0<A<15)

5. A=ASC(A$)-48+(ASC(A$)>64)*7 changes hex A$ to dec A (where "0"<A$<"F")

6. POKE56325,x changes the number of interrupts per second. This means that cursor flash speed changes, repeat keys repeat at different speeds, TI is inaccurate, program speed changes. This can be used to change the speed of interrupt driven music.

Also many thanks to you, Peter, for sharing that information with the other readers. Remember, if any of you have tips for publication, please send them to us at the usual Techno-Info address.

WHERE TO WRITE

If you are experiencing any computer-related problem, or you simply wish to air your views or have a tip published, then please write to me, Jason Finch, at the usual address:

CDU Techno-Info
11 Cook Close
Brownsover
Rugby
Warwickshire
CV21 1NG

Please do not send your letters to the CDU office as this can result in a delay in you receiving a reply or having your letter published. Thanks - see you all again next time.

LITE MENUS! (menu creation system)

Create professional looking menus with the minimum of effort. - Madhu Surendranath.

LITE MENUS! was created to take the effort out of menu-based programming and to create impressive results. It is written in BASIC with a few SYS calls to the Kernal ROM, however, it will work on any Commodore 64 (version 1 included as I own a Version 1 Commodore 64 and I have tested the routine on it.) The routine is built up of 3 separate routines. These are;

1. BOX ROUTINE - To add borders to the menus if required : (LINE 10000-).
2. MENU MAKER - The routine which adds a hi-light bar to the menu and all the selection processing is made there : (LINE 11000-).
3. PRINT AT - To move cursor to (X,Y) positions : (LINE 12000-).

HOW TO USE EACH SUBROUTINE.

BOX ROUTINE - The box controlling variables have to be set. They are :-

X - Move cursor to the right by X (0-39).
Y - Move cursor down by Y (0-24).
X1 - Sets width of the box (0-39).
Y1 - Sets height of the box (0-24).
CH - Selects character for box from the list below :-

CH=1 - character used (CBM + keypress).
CH=2 - character used (CBM A keypress).
CH=3 - character used (SHIFT U keypress).

The above list denotes the characters used for the corners of the boxes. If you select CH=1 then the box will be made up of a hatched pattern, CH=2 the box will have square corners and if CH=3 the box will have rounded corners.

COL- Selects colour of the box (0-15 Standard Commodore Colours).
So if you define all the variables, you need to call the subroutine with GOSUB 10000

Example definition of a box.

```
X=5:Y=5:X1=10:Y1=5:CH=1:COL=4:GOSUB 10000
```

This will print a box at position (5,5) and of size (10,5) of the hatched pattern and of colour PURPLE.

MENU MAKER - An array and variables have to set before use. They are :-

DIM O$(X) - The array which LITE MENUS! uses has to be defined as O$(X). X denotes the number of titles the array can hold : (between 1 and 23 usually)

```
O$(1)="OPTION 1"
 .
 .
O$(4)="OPTION 4"
```

Next, all the subject titles have to be set up, as shown above. It is stored in standard array format. Now controlling variables have to set.

MIN - First number of the array list (usually set to 1).

MAX - Last number of the array list (depending on array size).

PS - Number of lines from the top of the screen -1 (Y-1).

A - Control variable (to ease discrepancies, set to 1).

TA - Distance from left hand edge (same as X).

Once all these variables have been defined, you call this subroutine with GOSUB 11000.

Example definition of setting up a menu.

```
DIM O$(5)
O$(1)="OPTION 1" : O$(2)="OPTION 2" :
O$(3)="OPTION 3"
O$(4)="OPTION 4" : O$(5)="OPTION 5"
X=10 : Y=10 : GOSUB 12000 (**)
FORJ=1 TO 5 : PRINT TAB(X) O$(J) : NEXTJ
MIN=1 : MAX=5 : PS=9 : A=1 : TA=10 : GOSUB 11000
```

(**) This line moves the list to the positions as defined at (X,Y).

Depending on how you use this routine, there will be an extra line :-

ON A GOTO ...

As an option is selected, the choice number is stored in variable A so you can have access to this number if you need to.

PRINT AT - is simply what it says.

Continued on Page 47

Sacram

STUART ALLEN gives you an insight into a future world, or is it already here???

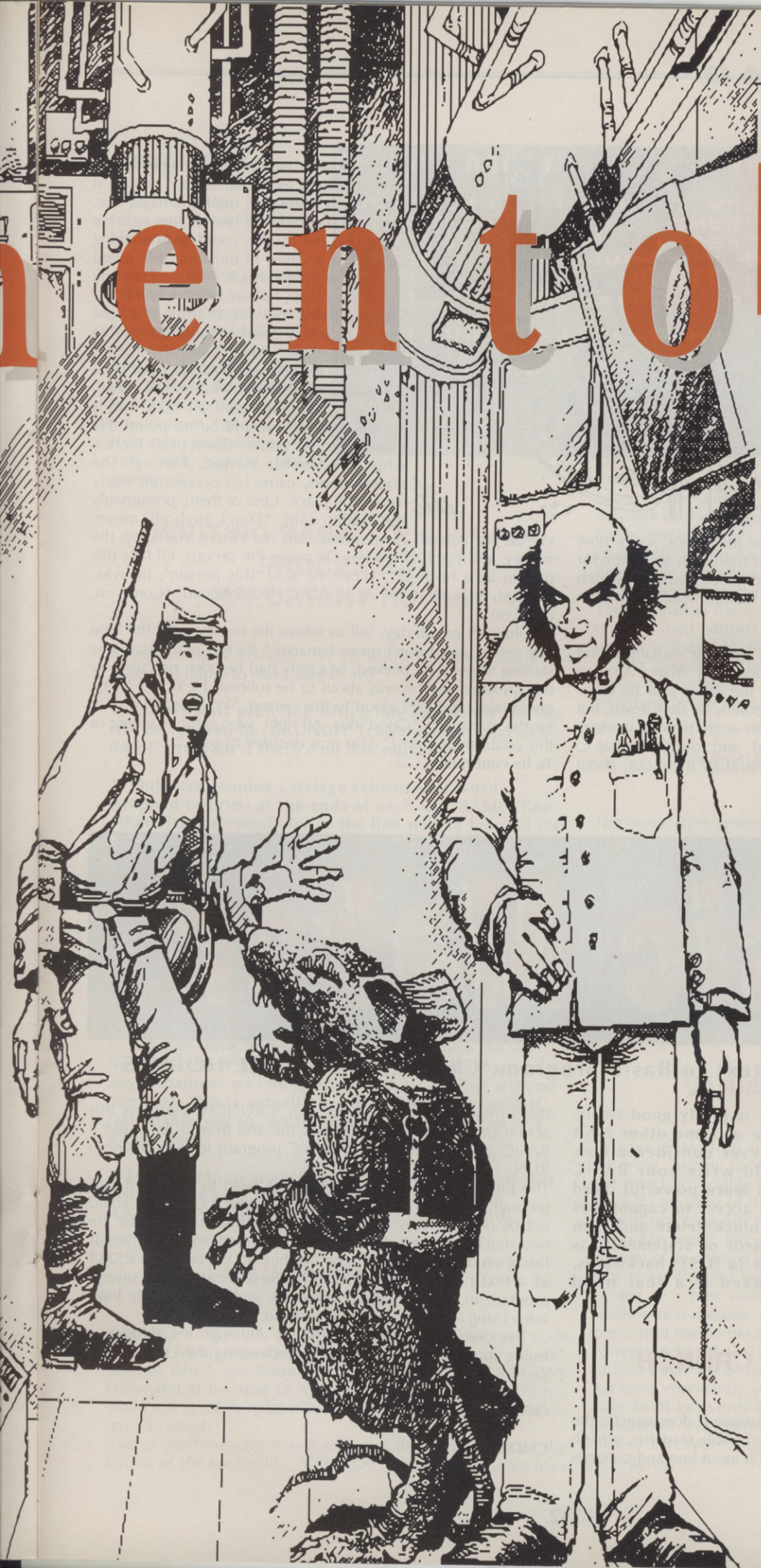
It was the year 781-70.

After the nuclear holocaust, the world was barren and drab; the surface everywhere had been discoloured to a dingy brown, the trees had no leaves signifying life, they were, themselves, scorched and black. There was one thing, one thing only, that to the visible eye showed life, that was something that looked like a hairy hand, clinging to a wheel. It was being held on by so little metal, the metal was imperceptible to anyone, or anything, unless they went up close and examined it. Another hand came up slowly towards the wheel, nearly got hold of it, then fell. A couple of minutes later, it tried again. It put up so much of an immense struggle to grasp the wheel, that when the hand did finally get hold of it, all the small strands of metal broke. The wheel fell, with the hands still clinging over it for dear life. There was a soft, dull thud, a yell, a few curses, then the hands sent the wheel hurtling into space.

From down the road, there was something going extremely fast, blurring the vision of it slightly. It slowed down to a stop, five seconds after it had seen the wreck of a car, with a body lying next to it. The driver undid his seat belt and got out. He did not look like the type to own such a wonderful vehicle. He walked calmly over to the person (who was now struggling to get up) and offered his assistance. It was definitely gladly accepted. After the various diagnosing and curing of wounds was done, the two introduced themselves, and delved into each others dark and murky past. Apparently, the driver was called Alan Mark Frederickson, he was somewhere in the region of sixteen to seventeen; he had also got two older brothers, one called Andrew Joe, the other called Ben James. He had a younger sister, Anne Marie. Both of his parents were killed in the holocaust. He used to be a military demolitions disposals man. That is, until he found a strange yellow substance, which changed him both physically and mentally. After that incident, the military disassociated themselves with him, and gave him a hefty wad of money, on the condition that the story never got out into the paper. From the substance, his memory increased, he was capable of learning things in a matter of minutes, instead of years. He was also changed from a Prairie Dog, to a mutated human. He was a dangerous man, judging by what his pastimes were. He liked to work-out in his private gym. When he wasn't doing that, he did some road scrambling in his favourite car which, incidentally, is what he was driving, when he saw the accident. If he ever got bored doing either of those two activities, he sometimes went doing target practice with his favourite sub-machine gun.... on real targets. He didn't tell much more about what his life was like, so he prompted the other man to tell. He was called Doctor Victor Oban Feral. He had a twisted imagination before the shock of people trying to assassinate him. Once he found that out, he somehow became more dignified, reserved, refined and utterly sophisticated. He would never stoop to insults or rude remarks and would be unfailingly polite, even to vivisection victims. He didn't allow racial or sexual discrimination in any way, shape or form, in any of his vast empire of businesses. He condemned dictatorship or racist remarks or behaviour, he was generous with friends, employees, educational institutes and the poor. In short, he was the ideal citizen, he believed weapons were for sport only. They chatted merrily for about half an hour. When Alan was just about to finish a sentence, he glanced down at his watch. "Oh, god!" "What's up?" "It's half past eight, I've got to get to Tipton for ten to." "Why?" "I've got to open the shop." "I've always been interested in how they open shops. Mind if I come along?" "Oh no. You can come. Quick, though, get into the car." Alan raced over to his car. The doctor remarked on it being very well protected. Alan replied by telling him that he hadn't seen the best of it. Alan literally stepped on it, to get the car going. He was going very fast, in fact he was going so fast he was breaking the sound barrier. When they got to Tipton. Alan said, "You see that field of grass?" "Yes" "Well, that's where I'm going to set up shop." "I thought you meant a proper shop" "Just you wait" Alan put his hands to his temples and seemed to be thinking. To the doctors great surprise, the field sank a couple of inches, then slid away.

"Oh, Rat's!. It's half past eight, I've got to get to Tipton for ten to."

A huge building appeared, slowly rising through the earth. The doctor said "Impressive." "You haven't seen the half of it" They waited for the building to rise up fully before taking in the full beauty of it. According to the doctor, there was just one small problem with it. He was getting a little bit worried about it, so decided to ask Alan about it. He said, "Errrm.. Alan, just exactly how do you get into it?" "Through the door, of course." "What door? All I can see is a huge tower, covered in steel." "That's where I come in again." At that moment, Alan put his hands to his temples again, and the sheets of steel folded away, into the building. The Doctor, although thoroughly perplexed by what appeared to be the tenth wonder of the world,



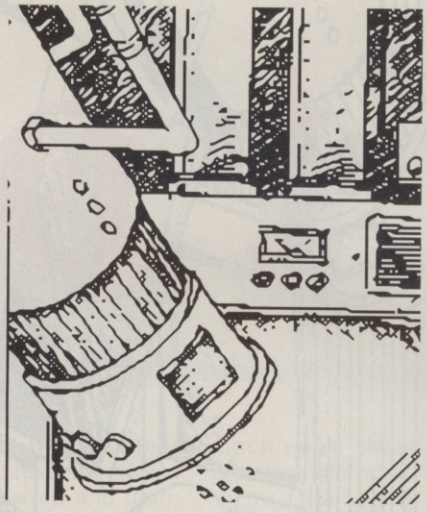
TRANDIMENSIONAL Teenage Mutant Ninja Turtles, by Erick Wujcik.. Published by Palladium Books.

have a hell of a lot of money." "Well... I should admit that I have a bit stashed away ", Then he muttered under his breath "He said, modestly." "I said that because I have got a small proposition to make to you, and wasn't sure whether you could afford to take it on.

You see, I've got something, back home, that isn't assembled properly, which will need plenty of spare space, just lying around. That is... if you're interested?" "I could tell you if you actually got to the point. Just what are you trying to ask me?" "If you could house a machine that would be able to earn you more than you ever dreamed. And no, it isn't one of those 'get rich quick' schemes. What it would be able to do, if I could get the space, is to create a copy of sections of a persons or things life. Namely the educational years which they amassed during their time in the great outdoors. It then puts that copy into a store, ready for anybody, or anything to use that version. Are you interested?" "That depends on how much you want for it." "Well, all I want is something for my time spent in making the machine, and getting rid of the bugs. Just what do you do in your shop?" "We make cars and weapons. We also sell them at two thirds of the normal price" "Nice. Right, let's say three dollars for every hour I spent making it, and you transport it here." "It's a deal. Just where is it?" "On route eighty. You know Sacramento?" "Yes" "Well, it's just off that." "I think I know where it is. I'll collect it tonight. Why don't you stay here until then." "That there's a clever person, here thinking of me staying here.

I am feeling a little bit sleepy. Is there anywhere I can sleep?" "Yes, just follow me." Victor was led through the rooms in the complex, into a room which Alan called, the lift. It was extremely lush, with old masterpieces decorating the otherwise sparse walls. As he was examining them, a soft voice came over the loudspeakers and asked which floor they wanted to go to. He heard Alan mutter 'third'. There was a whirring, which stopped almost as soon as it had begun. Alan, being the cheerful chap that he was, told the Doc., "Here we are. Third floor, everybody out, women and old ladies first", looking at the doctor as he said

'old ladies'. He then continued to say, "Children get thrown to the back before the rush, Try not to ruin the carpets with the wheelchairs, you older ones, and most of all: Keep the pets under control. Preferably under the owners control." "Just what did you mean by glancing at me when you said 'old ladies'?", queried the doctor. "I had to look somewhere", said Alan, with a smile. "Ah. But why did you look at me?" "Well, the paintings would've complained." "All this mind-boggling conversation has given me a slight headache." grumbled the doctor, "We don't appear to have moved." "Can you remember roughly what the outside of the lift looked like, before you came inside? If you can, have a look back out, and see if it looks anything at all like what you remember." He did so and to great surprise found that they had, in fact, moved. Alan gave the doc. a quick look around the third floor, so he now knew where the first aid room was, and snook out an aspirin for his headache. He then entered the gorgeous bedroom which Alan used for his guests, and was told that he would be woken up at around five.



It was now about ten past nine. Alan was just walking round the complex when he heard several screams of 'Alan... Are you in there?'. He had forgotten to unlock the doors after he came in. He rushed back downstairs (well, actually he flew down, but that little point is irrelevant), and met his angry team of workers yelling at him. He rapidly apologized, and told them that to make up for the inconvenience, he would let them out, about fifteen minutes early. They all agreed, and so they began to set up shop. Overall, it was a fairly quiet day, until quarter to five. Alan was pleased with himself, because He had found out that he could do something strange, as a cause of mutation. He found out that he could shrink and grow, to any size, at any time, for any length of time. He was just trying this out, in front of all the staff, after telling them about it (the people who work there, never have any secrets from Alan, for two reasons. One, that he is a friendly boss, who would not tell a soul. And two, he could read all their minds anyway, so there would be no point). He managed to get himself down to six inches, when the trouble started. Through the double doors, came ten people, all ready to rob the place. One of them, presumably the leader, said, "Don't anybody move. We've got you all surrounded. Tell me where you keep the money, and you'll all go free. To prove I'm serious, I'll take this person as a hostage". When he said 'this person', he was, strangely enough, referring to Alan. He continued, "Come on, come on.

We haven't got all day. Tell us where the money is, or the little guy gets it, and I don't mean bananas." By this time, Alan was starting to get a bit peeved. Not only had he been ridiculed by this creature, but he was about to be robbed. He'd had all he could take, and said aloud to this person, "Why don't you ask for the manager?" "Good idea, All right, who is the manager of this establishment, eh?" Alan now decided to grow.....

To be continued.....

Basic Maker

At last a C64 ASCII text to Basic program 'CRUNCHER' - MIKE HOLMES

The dear old C64's BASIC editor is pretty good really, when you compare it with those of some other 8-bit machines. However, have you ever considered how useful it would be if you could write your BASIC programs using something like a more powerful word processor? You would then have access to capabilities like search/replace, copy/move, block delete and even defined 'macros' for oft used words or statements, as well as the novelty of being able to 'LIST' backwards. Well now you can. All you need is a real word processor, and this program.

IT'S ALL IN THE CRUNCH

It is called 'CRUNCH', because in essence it does exactly the same as the BASIC interpreter's crunch routine, which recognises BASIC keywords in an ASCII input line and converts them into single byte tokens. But 'CRUNCH' performs this action on a complete ASCII disk file and from it generates a BASIC program disk file in BASIC program format, ready to RUN.

The program was developed out of a need to transfer BASIC programs back and forth between a PC and a C64. This could not be done directly for all sorts of reasons, not least that the two machines are totally different. Instead an ASCII format listing on disk is sent from one machine to the other via RS232 as a text file. The GW-BASIC of the PC can easily save a program in ASCII format if required, and just as easily load same using a special 'merge' command.

The Commodore however cannot. Although it CAN save a listing to disk in ASCII format by redirecting the LIST output, i.e.:

```
OPEN 2,8,2, "BASLIST,W,S"

CMD2 : LIST
```

CLOSE2

which is easy, it cannot reload and run the result. Until now.

Apart from the increased editing power of the word processor, something else becomes apparent. The C64 will actually accept BASIC lines in memory as long as you like, but the problem is that while using the editor you cannot type in more than two screen lines. But you can on a word processor.

```
Crunch ASCII file to BASIC prog.

Source file must be written in BASIC
format with line numbers etc., and
must be pure ASCII text only.
Lines without numbers will be ignored.
Product will be BASIC prog. with
filename extension ".bas".

Enter ASCII file name : ■
```

WRITING BASIC ON A WORD PROCESSOR IMPORTANT RULES

Before using 'CRUNCH' on an ASCII file there are a few important things to make sure of first.

1. **Always start each line with a line number or the line will be ignored by 'CRUNCH' (can't make it part of a BASIC program if there's no line number).**

2. **Only use genuine carriage returns (not word wrapped returns) at the ends of each full 'BASIC' line, otherwise the remainder of the line will be ignored (no number at start). This presupposes that your word processor only saves returns at the ends of paragraphs etc., i.e. where you want them.**

3. **Don't use '?' for 'PRINT', but the whole word. Similarly don't use other 'short-hand' methods of writing keywords, which must appear in full.**

4. **Always make sure that line numbers are not out of sequence or duplicated etc.**

5. **You can have up to 255 characters in one 'BASIC' line. 'CRUNCH' can only input 255 in one lump, and any remainder will be ignored (like a new line with no number). This is actually a useful amount and will at last enable you to 'get the last bit in' on the end of a BASIC line.**

6. **It may help to always use spaces between keywords just to make sure there's no confusion, although 'CRUNCH' should cope OK with closed up words.**

USING 'CRUNCH'

Type 'LOAD "CR",8', 'RUN'. 'CRUNCH' is a machine code program which uses a sort of standard routine library 'rtl.exe', both of which are loaded in by 'cr'. There is a BASIC SYS command at the start of 'CRUNCH' so that it can be re-run while still in memory from BASIC direct mode (but don't create any variables!).
After a short message which reiterates the warnings about the format of the source file, 'CRUNCH' asks for the source file's name. You now put in the disk with the file on it and enter the name. An invalid name causes the display of the DOS error message and ends the program.To create the destination file name, the source file name is truncated to a maximum of 12 characters and '.BAS' is added to the end. If the source file name had '.'+<something> on the end it is replaced with '.BAS' for the destination name. This is then displayed for reference. The destination file is always type 'PRG'. The file starts with a dummy load address, and each BASIC line with a dummy pointer followed by the line number as an integer, and ended with a zero byte. The dummy numbers are not a problem; when the completed program is loaded normally these are sorted out by the interpreter and given their proper values.

KEYWORD HUNTING

The translation process then commences. Each text line is scanned and converted to BASIC format, then written to the destination. Each line is printed on screen to show progress, together with what appears to be a counter at the bottom left hand corner.
'CRUNCH' has a reference vocabulary of valid C64 BASIC keywords. On matching one of these words to a word in the line the BASIC token is displayed in decimal while the token is sent to the destination instead of the word. The display of this token is the 'counter'.Everything after a 'REM' is ignored and copied straight to the destination, similarly, everything between quotes (" ").

VARIABLE LEARNING

To save execution time, 'CRUNCH' learns to recognise recurring variable names. Instead of laboriously failing to not just match a variable to a standard keyword, but every letter thereof, an array of these words is built up and used like a second matching list. The display of a '*' at bottom left signals a variable match, followed by - if you have time to see it - the number of its position in the variable list.
This is very useful because at the end all the variables are printed on screen with their type ('%' or '$') if applicable. In addition non-numeric data items are shown. (Alphabetic data items must not be the same as BASIC keywords or they will be converted to tokens!) You will immediately see all of the variables that will be used by the final program.
But more importantly, you may see 'funny words' which should have been keywords but were misspelt. The program file is terminated with two zero bytes (end of BASIC prog.) and closed. 'CRUNCH' then returns control to direct mode, but can be restarted with 'RUN'.

ON THE WINDOWS C64

Using windows
on your C64
couldn't be easier!
- F.E.RANDALL

This program provides all the facilities you'll need to create a window environment for your Basic programs. These routines can also be used by means of SYS calls typed in from the keyboard. They allow you to specify the size and shape of the windows you require.

TECHNICAL BITS

When a window is invoked, all the normal screen editing functions are available, but they only operate on the area of the window you have specified. When that window is 'PUSHED BACK', the original screen is restored. Up to four windows may be defined, and each may be 'PULLED DOWN' in whatever order you determine.

The definition of a window includes its position on the screen, but when 'PULLED DOWN', it may be 'DRAGGED' to another position, and it then becomes the new location for that window in subsequent operations.

The areas used by the system are $C000 to $C830 for the machine code, and the screen data is saved under Basic RAM at $A000 to $BFFF. The locations $F9 to $FE are used for the parameters of the current window, and must not be disturbed whilst the window is 'PULLED DOWN' otherwise the system may crash. The original values in these locations are saved each time a window is 'PULLED DOWN', and restored when the last window is 'PUSHED BACK'.

To achieve the usual screen editing functions whilst only operating on the area of the window, the system includes rewritten parts of the Kernal routines CHRIN and CHROUT. When a window is extant, the vectors at $0324 and $0326 are changed to $C300 and $C000 respectively. Pressing RUNSTOP/RESTORE will reset these vectors if you run into difficulties.

HOW TO USE THE SYSTEM

To use the system, you first have to initialise it by SYS50176. That call should also be used if there has been an error message, since the parameters can be in an indeterminate state after such an event. More about the error messages later.

After initialisation the window have to be defined by SYS50179,A,B,C,D,E where;

A = The window serial number from 1 to 4. This number is used to pull down the window later.

B = The number of the row on the screen where the top left and corner of the window is to appear.

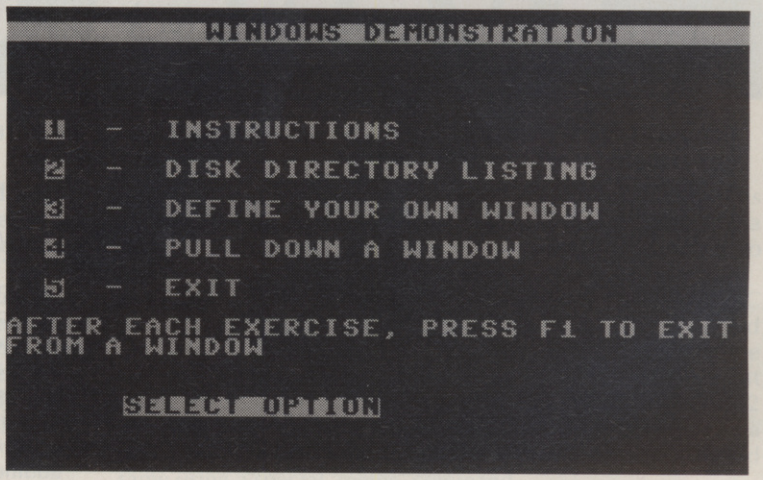
C = The number of the line on which the top left hand corner is to appear.

D = The width of the screen in characters.

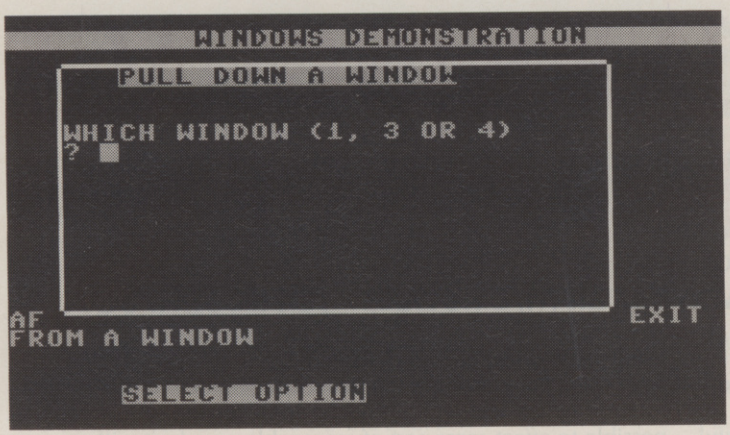
E = The number of lines, ie the depth of the window.

The window must be a minimum of three characters wide and three lines deep, and the starting row and line plus the width and depth must not exceed 39 and 24 respectively, since these are dictated by the screens dimensions. In practice, you'll want to leave ample room for any Basic commands you wish to enter from the window.

This brings me to the one exception to the normal screen editing facilities which the system imposes. Normally there is a wrap around on input so that each line may be up to 80 characters long. In this system the input line is restricted in length to the width of the window you specify. You must make allowances for this when setting the parameters.


WINDOWS DEMONSTRATION
1 - INSTRUCTIONS
2 - DISK DIRECTORY LISTING
3 - DEFINE YOUR OWN WINDOW
4 - PULL DOWN A WINDOW
5 - EXIT
AFTER EACH EXERCISE, PRESS F1 TO EXIT FROM A WINDOW
SELECT OPTION

The definition of windows will probably come in the initialisation procedures of your program. If you're going to use them to display preset messages, you will probably also want to set up the displays in the initialisation by 'PULLING DOWN' the window, PRINTing the text, then 'PUSHING BACK' the window. 'PULLING DOWN' is achieved by SYS50182,A where



'A' is the window number as used in the definition. SYS50185 will 'PUSH BACK' the window.

Windows can be redefined without re-initialising the system, but once redefined, the original contents of the window will be lost. Furthermore, if re-definition takes place too frequently, you may run out of space in which to store the contents of the windows. Better to use the same window for different purposes, and clear it by PRINTing "CLS" each time.

DRAGGING WINDOWS

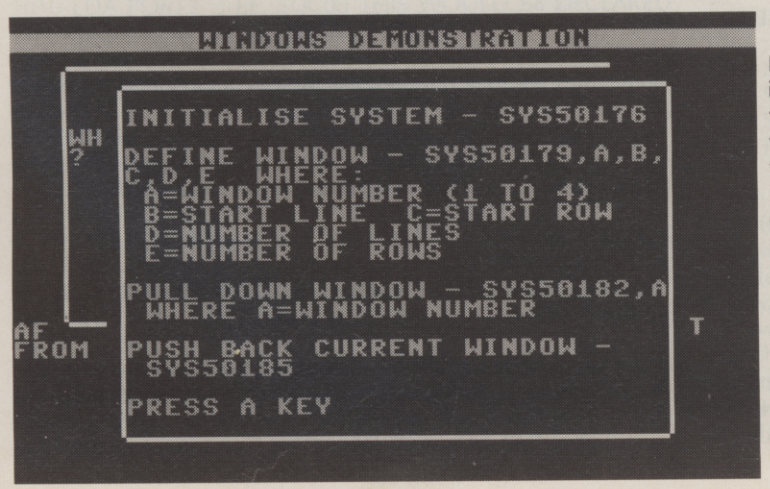
Having 'PULLED DOWN' a window, you may drag it round the screen using the following commands;

UP SYS50191
DOWN SYS50188
LEFT SYS50894
RIGHT SYS50197

To drag the window from your Basic program, you would probably want to test for the pressing of a particular function key and then use the appropriate SYS call. For more flexibility, machine code buffs could 'WEDGE' code into the interrupt routine to check whether a function key is pressed and if so, call the relevant subroutine listed above.

ERROR MESSAGES

The system generates error messages as shown below to help in debugging your program. The conditions are mostly related to the use of incorrect parameters.



Since the system may be left in an indeterminate state when the error is detected, after displaying the message, the system waits for a key to be pressed before carrying out a warm start. It's always best to initialise the system again after such an occurrence.

The exceptions to this are the absence of parameters following the SYS commands, as this is picked up by the Basic interpreter and results in a SYNTAX ERROR message. The message displayed will take the form "ERROR -" followed by a letter. The significances of the letters are as follows;

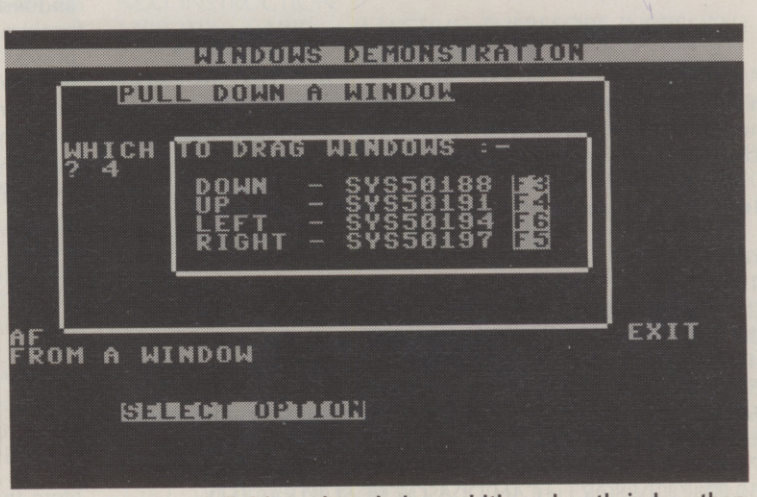
A - An attempt made to 'PULL DOWN' a window before it has been defined.

B - An attempt made to 'PULL DOWN' more than four windows or the same window.

C - When defining a window either the start row plus width exceeds 39, or the start line plus length exceeds 24, or there is no more space to store the window contents.

D - In a window definition, the window number is not in the range one to four.

E - When 'PULLING DOWN' a window, the window number is not in the range one to four.



F - In a definition, the window width or length is less than three.

DEMONSTRATION PROGRAM

Included on the disk is a demonstration program introducing the use of window. Load and RUN "WINDOWS DEMO". To use the machine code in your own program type LOAD"WINDOWS MC",8,1 then NEW. The demonstration gives you the choice of 'PULLING DOWN' windows containing instructions on how to use the system; using a window to view the directory of a disk; defining a window of your own; or 'PULLING DOWN' one of the windows used in the program. The latter uses window number two, so if you try to 'PULL DOWN' that window, you will get 'ERROR - B'.
If you study the listing of the demonstration program, I'm sure you will soon find all is made clear. You may also wish to incorporate the directory listing routine in some of your own programs. Happy Windowing!!

BASIC MACHINE LANGUAGE TECHNIQUES

Part Four of this series gets underway for all M/L novices - JOHN SIMPSON

As with BASICS OF BASIC, we had to omit last months offering of BASIC MACHINE LANGUAGE TECHNIQUES, I hope it did not mar your enjoyment of the magazine.
So far we have covered many of the aspects towards successful programming in Machine Language (or Assembler!). This month we shall continue by discovering some of the aspects in program construction and development.
There are as many ways to construct a program as there are people constructing programs - what I mean by this is that each programmer will usually develop his or her own methods, shortcuts, and design. Mine is but one of this myriad of techniques and as you develop your own programming skills then your own particular method will shine through.
During this phase of the series we will construct routines which can be used in the different programs that you may develop in the future. These can be separated and saved to a 'library of subroutines' for your future use.

BUT FIRST

In JUNE I left you to construct a division algorithm based upon the topics we covered with relationship to multiplication - did you manage the task to a successful conclusion?

Here is an example of a possible division program:

```
100 DIVISION
110 LDX #8        ; NUMBER OF BITS IN DIVISOR
120 LDA DIVIDEND  ; START WITH LSBYTE OF DIVIDEND
130 STA QUOTIENT  ; STORE IN QUOTIENT REGISTER
140 LDA DIVIDEND+1 ; GET MSBYTE OF DIVIDEND
150 DIVIDE
160 ASL QUOTIENT  ; SHIFT LSB'S OF DIVIDEND QUOTIENT
LEFT 1 BIT
170 ROL A         ; SHIFT MSB'S LEFT AND ADD CARRY BIT
180 CMP DIVISOR   ; CAN DIVISOR BE SUBTRACTED
190 BCC NOSUB     ; NO, SO GO TO NEXT STEP
200 SBC DIVISOR   ; YES, SUBTRACT DIVISOR (CARRY = 1)
210 INC QUOTIENT  ; AND INCREMENT QUOTIENT BY 1
220 NOSUB
230 DEX ; LOOP UNTIL ALL 8-BITS HAVE BEEN HANDLED
240 BNE DIVIDE
250 STA REMAIN    ; STORE REMAINDER
260 RTS
280 DIVIDEND BYT 10,10 ; LO/HI BYTE OF VALUE
290 DIVISOR BYT 10
300 QUOTIENT BYT 0
310 REMAIN BYT 0
330 END
```

MOVABLE OBJECT BLOCKS (SPRITES).

This month we are going to create a small program which will allow us to manipulate a Sprite. It will do nothing more than move the Sprite around the screen and change it's colour by using the joystick, plugged into Port 2,
After using Basic routines to manipulate Sprites within programs - routines which you will, no doubt, have streamlined as much as possible to effect some sort of reasonable speed - it will come as a pleasant surprise, perhaps, to know that performing the same functions using ML you will be striving, not to speed things up but, rather, to slow things down.In the development of these routines we shall need to produce some ancillary functions to enable us to test the routines. The first of these will be an efficient, all purpose, Joystick Routine, which could then be included into your library of subroutines.

UNIVERSAL JOYSTICK ROUTINE.

There are many ways to 'read' the joystick, each relevant to the program you may be developing at the time. However, a tried and tested routine which was first put together by 'Bill Hindorff' proves to be very satisfactory for most applications. If you possess a copy of "The Programmer's Reference Book" you can find the origination of the routine on page 345.
In this routine we will access the 'bit' information of the register used by the joystick plugged into Game Port 2. The register appears at memory location 56320 ($DC00), and it is the lower five 'bits' which are affected by joystick operation.

```
BYTE $DC00
      128 64 32 16 8  4  2  1          - BIT VALUE
      ---------------------
      7   6  5  4  3  2  1  0          - BIT NUMBER
      |   |  |  |  |  |  |  UP         }
DON'T CARE |  |  |  |  |  DOWN         }
           |  |  |  |  LEFT             }JOYSTICK OPTION
           |  |  RIGHT                  }
           FIRE                        }
```

Note that when a Joystick direction or the Fire button is NOT activated the relevant bit is set. This is done automatically by the processor as it performs its 'housework' during each interrupt sequence.

A SLIGHT DIVERSION

Sometimes we may only require joystick information for one or two events, such as 'FIRE' or 'LEFT/RIGHT'. This can be ascertained quickly by using a 'mask and compare' method. The first stage of this method is to mask out the three upper bits of the register which are not relevant to the joystick read, thus:

```
100 LDA $DC00      ; TRANSFER GAME PORT BIT INFO INTO
ACCUMULATOR
110 AND #%00011111 ; 'AND' IT TO DISPOSE OF THE UPPER
THREE BITS
```

Depending if the joystick has been activated, then one or more of the subsequent 5 bits will be cleared (= 0). For example if the fire button has been depressed then the bit pattern of the Accumulator would register '00001111', or 15 (bit-4 having been

cleared), so:

```
120 CMP #15      ; COMPARE FOR FIRE BUTTON ACTIVITY
130 BNE SKIP     ; IF IT IS NOT EQUAL TO 15 THEN BRANCH TO SKIP
140 JSR FIREACTION ; YES IT DOES EQUAL 15 SO CALL A SUBROUTINE TO DEAL WITH A FIRE BUTTON ACTIVE SITUATION
150 SKIP
160 ; *** REST OF THE PROGRAM ****
```

You will quickly spot that you can easily compare the Accumulator with other various values to determine which direction the joystick may have been moved:

```
160 CMP #30      ; 00011110-UP-IS BIT 0 CLEAR?
170 BEQ UPACTION  ; YES SO BRANCH
180 CMP #29      ; 00011101-DOWN-IS BIT 1 CLEAR?
190 BEQ DOWNACTION ; YES SO BRANCH
200 CMP #27      ; 00011011-LEFT-IS BIT 2 CLEAR?
210 BEQ LEFTACTION ; YES SO BRANCH
220 CMP #23      ; 00010111-RIGHT-IS BIT 3 CLEAR?
230 BEQ RIGHTACTION; YES SO BRANCH
240 CMP #22  ; 00010110-UP/RIGHT-IS BITS 0 AND 3 CLEAR?
```

and so on. As you can see, to 'capture' every combination of joystick activity would require some 17 compares (that is the eight directions without the fire button depressed, and another eight with the fire button depressed, plus the fire button alone). However using the 'mask and compare' method does have its advantages when you may want to force the user to operate the joystick in a certain manner - e.g. fire or left/right or up/down only etc.

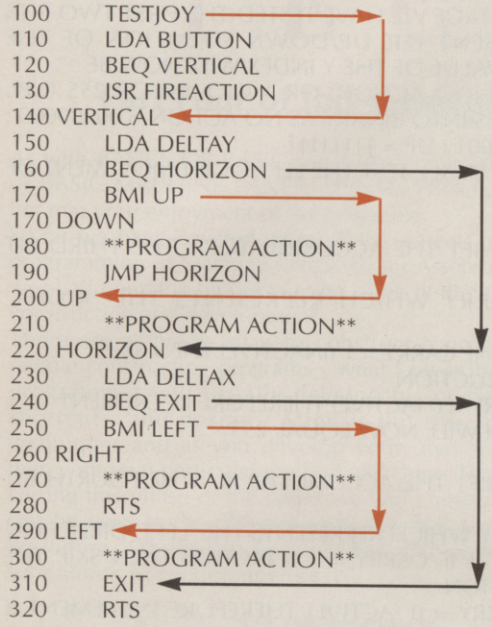
BACK TO THE UNIVERSAL

Returning to the universal routine of 'Bill Hindorff'; this routine does not use a mask but instead allows us to test each of the joystick bits by using right shifts (LSR) and reading the carry flag after each shift. We then adjusting the X and Y Indices accordingly and finally place the results into three variable data locations which we can call DELTAX, DELTAY, and BUTTON.
First we need to read the relevant joystick port register and copy its content into the Accumulator. We will then zero the X and Y indices together with the BUTTON variable data byte. After this it is a simple matter of performing consecutive right shifts on each of the first five bits held in the accumulator's copy of the port register which will comprise of the current joystick status. As we shift each bit out of the Accumulator into the Carry bit we know that if the bit is set, then that particular joystick option has not been selected. So, the start of our joystick routine is thus:

```
1000 JOYSTICK
1010 LDA PORT2  ; I.E. $DC00 THIS WOULD BE DECLARED WITHIN
OUR EQUATES AT THE PROGRAM HEADER - WE DISCUSS EQUATES SHORTLY
1020 LDX #0     ; ZERO THE X AND Y INDICES AND BUTTON DATA BYTE
1030 LDY #0
1040 STY BUTTON
1044 ; THIS COMPLETES THE ROUTINE INITIALISATION ***
1050 LSR A  ; SHIFT THE ACCUMULATOR - THE FIRST BIT DROPS
INTO THE CARRY WHICH REPRESENTS THE 'UP' DIRECTION
1060 BCS SKIP1  ;IF CARRY = 1 (INACTIVE) THEN SKIP THE NEXT INSTRUCTION
1070 DEY ; CARRY = 0 (ACTIVE) THEREFORE DECREMENT THE Y INDEX WHICH WILL NOW EQUAL 255.
1080 SKIP1
1090 LSR A ; SHIFT THE ACCUMULATOR-THE SECOND BIT DROPS INTO THE CARRY WHICH REPRESENTS THE 'DOWN' DIRECTION
1100 BCS SKIP2 ; IF CARRY = 1 (INACTIVE) THEN SKIP THE NEXT INSTRUCTION
1110 INY ; CARRY = 0 (ACTIVE) THEREFORE INCREMENT THE Y INDEX
WHICH WILL NOW EQUAL 1.
1122 ; AT THIS STAGE WE HAVE TESTED THE FIRST TWO BITS WHICH REPRESENT THE UP/DOWN SITUATION OF THE JOYSTICK, THE VALUE OF THE Y INDEX WILL NOW BE
1124 ; EITHER 0 (NO ACTION) OR 1 (DOWN) OR 255 (UP). THIS TRANSLATES INTO BINARY AS NO ACTION = 00000000 : DOWN = 00000001 : UP = 11111111
1126 ; NOW WE SHALL TEST THE LEFT/RIGHT MOVEMENT OF THE JOYSTICK.
1130 SKIP2
1140 LSR A ; SHIFT THE ACCUMULATOR - THE THIRD BIT DROPS
INTO THE CARRY WHICH REPRESENTS THE 'RIGHT' DIRECTION
1150 BCS SKIP3 ; IF CARRY = 1 (INACTIVE) THEN SKIP
THE NEXT INSTRUCTION
1160 DEX ; CARRY = 0 (ACTIVE) THEREFORE DECREMENT
X INDEX WHICH WILL NOW EQUAL 255.
1170 SKIP3
1180 LSR A ; SHIFT THE ACCUMULATOR - THE FOURTH BIT DROPS
INTO THE CARRY WHICH REPRESENTS THE 'LEFT' DIRECTION
1190 BCS SKIP4  ; IF CARRY = 1 (INACTIVE) THEN SKIP THE NEXT INSTRUCTION
1200 INX ; CARRY = 0 (ACTIVE) THEREFORE INCREMENT X INDEX WHICH WILL NOW EQUAL 1.
1212 ; AT THIS STAGE WE HAVE TESTED THE SECOND TWO BITS WHICH REPRESENT THE UP/DOWN SITUATION OF THE JOYSTICK, THE VALUE OF THE X INDEX WILL NOW BE
1214 ; EITHER 0 (NO ACTION) OR 1 (LEFT) OR 255 (RIGHT). THIS TRANSLATES INTO BINARY AS NO ACTION = 00000000 : LEFT = 00000001 : RIGHT = 11111111
1216 ; FINALLY WE TEST THE FIREBUTTON.
1218 ;
1220 SKIP4
1230 LSR A ; SHIFT THE ACCUMULATOR - THE FIFTH BIT DROPS INTO THE CARRY WHICH REPRESENTS THE 'FIRE BUTTON'
1240 BCS SKIP5  ; IF CARRY = 1 (INACTIVE) THEN SKIP THE NEXT INSTRUCTION
1250 INC BUTTON ; CARRY = 0 (ACTIVE) THEREFORE INCREMENT THE VARIABLE BYTE, BUTTON TO 1
1262 ; WE HAVE NOW READ ALL THE RELEVANT BITS OF THE JOYSTICK STATUS AND HAVE PLACED THE INFORMATION INTO BOTH THE X AND Y INDICES AND THE VARIABLE
1264 ; BUTTON - ALL THAT REMAINS TO BE DONE IS TO UPDATE VARIABLES DELTAX AND DELTAY FROM THE INFORMATION HELD IN THE X/Y INDICES.
1266 ; 1267 ;
1270 SKIP5
1280 STX DELTAX ; TRANSFER X INDEX TO VARIABLE DELTAX
1290 STY DELTAY ; TRANSFER Y INDEX TO VARIABLE DELTAY
1300 RTS        ; RETURN TO CALLER.
```

When we wish to use the information just gathered we will know that BUTTON will equal 1 if the fire button was pressed otherwise it will be 0, that DELTAX will equal 1 if RIGHT was selected, 255 if LEFT was selected otherwise it will be 0, and finally, DELTAY will equal 1 if DOWN was selected or 255 if UP, otherwise 0. These values are very simple to test for conditional branches by using the BPL or BMI instructions.
We know that a zero indicates no action, and that UP or LEFT = 255 which means that the 7th Bit will be set but if DOWN or RIGHT have been selected then the 7th bit will be clear, so by using the BMI for LEFT or UP will cause a conditional branch.

Here follows a typical algorithm for testing and action upon the joystick information which has just been gathered.
Note the use of the BEQ and BMI which allows us a three-way switching technique - off/direction1/direction2:(The broken lines demonstrate the various program flows depending upon the action of the Joystick).



THE PROGRAM MAIN LOOP

When I write a program I usually create a MAIN processing loop and from within this loop, which determine the flow of the program, I can then easily access all the routines - it is from within MAIN where decisions on the result of computation or whatever are carried out and program flow is diverted or adjusted. This greatly aids in debugging the program, adding new routines to the program or deleting old ones, etc. Because MAIN ends with an unconditional jump back to its start I can easily exit the program back to basic, reset the computer. or JMP to ancillary or alternative programs.
For our sprite program we shall construct a simple MAIN which will allow us to test the routines which go to making sprites flow.

500 MAIN
510 JSR JOYSTICK ; OUR JOYSTICK ROUTINE FROM EARLIER
520 JSR SPRTMOVE ; A ROUTINE WHICH DETERMINES AND EXECUTES SPRITE MOVEMENT
999 JMP MAIN ; REPEAT MAIN AGAIN - AN ENDLESS LOOP SITUATION.

MAIN would be adjusted as more routines were constructed, such as randomly moving sprites, collision detection and action on collision, screen updates and etc. Now, however, we shall simply block in some line numbers to determine where the routines for this month will be located.

1000 JOYSTICK
1999 RTS
2000 SPRTMOVE
2999 RTS

SETTING THINGS UP

Before the program enters MAIN, however, we will need to 'set up' various things, such as clearing the screen, initialising data bytes, or sprite information, copying screen graphic data from storage to the actual screen, or what have you. So, before MAIN we shall call a subroutine named, SETUP.

100 JSR SETUP
3000 SETUP
3999 RTS

So far we have allocated various space within which we shall locate our routines and have made a start towards structuring the program design. As we develop each subroutine we can open up a gap within the space allocated and enter our lines of code. For example, the universal Joystick routine of earlier can be typed into the section JOYSTICK, ready to be used.

In the SETUP routine we will need to clear the screen of any data we do not require. A good method for doing this is to use one of the 64's built in ROM routines - the same one which you use from Basic when you use, PRINT CHR$(147). The ROM routine which conducts a 'print' has its start location situated at $FFD2 and is known as a 'kernal' ROM routine, its label is CHROUT. By exploiting this routine all we need to do is place the character, or CHR$ (code number), we wish printed into the accumulator and call the ROM routine CHROUT. This saves us the need to construct a "PRINT" routine - Commodore have already done this for us (incidentally, there are many 'built-in' ROM routines that we can exploit. As we require the use of such routines I will give an explanation of them and how best to access them).
In our small program the SETUP routine will only require 16 lines. Let us examine this:

```
3000 SETUP
3010 LDA #147 ; THIS IS THE CHR$ CODE TO PERFORM A CLEAR SCREEN
3020 JSR CHROUT ; THE KERNAL ROUTINE TO DO JUST THAT!
3030 LDX #63  ; WE NEED TO FILL 64 LOCATIONS WITH DATA TO DISPLAY OUR EVENTUAL SPRITE
3040 LDA #255 ; FOR NOW I HAVE CHOSEN, FOR SIMPLICITIES SAKE, TO USE A BLANK SQUARE
3050 LOOP
3060 STA BUFFER,X ; HERE WE FILL THE 64 LOCATIONS FROM THE ADDRESS BUFFER WITH THE VALUE HELD IN THE ACCUMULATOR (255)
3070 DEX ; DECREMENT THE X INDEX - UNTIL IT REACHES ZERO AND FLIPS TO 255 WE KEEP LOOPING  BACK TO REPEAT THE FILL
3080 BPL LOOP ; EVENTUALLY THIS WILL BECOME FALSE (WHEN X REACHES 255) AND SO WE DROP THROUGH TO THE NEXT LINE.
3090 LDA #13 ; STORE THE VALUE 13 INTO THE ACCUMULATOR - THIS IS THE SPRITE DATA POINTER OFFSET VALUE
3100 STA SDP ; AND TRANSFER IT TO OUR SPRITE DATA POINTER - THIS IS THE SAME AS: POKE 2040,13 IN BASIC.
3110 LDA #100 ; THE NEXT THREE INSTRUCTIONS WILL PLACE THE VALUE OF 100 INTO SPRITE ZERO'S
3120 STA VIC ; X COORDINATES AND
3130 STA VIC+1 ; Y COORDINATES
3140 LDA #1 ; HERE WE TRANSFER '1' (WHITE) TO SPRITE ZERO'S
3150 STA VIC+39 ; COLOUR REGISTER
3160 STA VIC+21 ; AND ENABLE SPRITE ZERO
3170 RTS ; THAT CONCLUDES SETUP SO RETURN TO CALLER
```

Unfortunately, space has once again beaten us. We will have to continue this part of our tutorial next month. Until then, brush up on all that we have learnt so far.

X-RAY FILES

Take a peek inside your 1541 and find out more about what's going on with our X-Ray program PAUL EVES

Users of the 1541 Disk Drive know already the advantages of this form of medium over the Cassette. Although the speed of the drive is something to be improved upon. (Although this is not the 1541's fault, but the operating system of the 64). However, not everyone uses their 1541 to its full potential. Indeed, most users of the 1541 don't know how the drive can be put to work for their own advantage. The program FILE X-RAY sets out to show you that with a little thought on your part, you can make programming the drive a pleasure and advantageous thing to do.

When you load the directory up, all it tells you is the name of the file and how many blocks it is, along with the program type. There are however, a few more bits and pieces of information stored in the Directory. Things like the starting address of a file. These other bits of information are known as the 'File Parameters' of a program. By understanding the make up of the directory better, these parameters can be easily found and printed to the screen or printer.

FINDING PARAMETERS

In total there are nine that we can unearth. They are:
1 File Closed?
2 File Protected?
3 Blocks Allocated
4 Side sector blocks (Relative files)
5 Data blocks (Relative files)
6 Records (Relative files)
7 Start Address (Program files)
8 Free blocks on disk
9 Allocated blocks on disk

The following program demonstrates how we find this information and then prints it to the screen of the 64 or to the printer.

Firstly, I have listed the variables used within the program, following this is the program breakdown. You should have no trouble following what's going on. The REMs may be omitted and of course you can alter the layout to suit your own personal taste.

Owners of the Vic can run the program by changing lines 16 and 17 (Screen colour set up).

VARIABLES

RE - Track of the file entry in the directory
SE - Sector of the file entry in the directory
AF - Lower 4 bits of file type (actual file type)
LO - Low byte of the start address
HI - High byte of the start address
TP - File type of requested file
NF - Not found flag if required. File on disk does not exist (set if not found)
BK - Number of blocks if file
FA - File address of program file (the start address)
LE - Length of record for relative file
DB - Track of Data Block of program file (contains start address)
DF - Sector of 1st data block of program file
FB - Free blocks on disk
AB - Allocated blocks on disk
SB - Side sectors in relative file
RF - Records in relative file
N$ - Name of requested file
FN$ - Directory file name
TY$ - File type
XX$ - Shows a closed or open file
SE$ - Shows a secure file or non secure (i.e. protected)

PROGRAM (LINES) BREAKDOWN

16-17 Set screen and border colours. Set cursor colour. Disable the LIST function. Disable SHIFT key. Disable RUN/STOP RESTORE. Clear screen.
18 Blank the screen.
19-25 Puts title screen up.
26 Switches screen back on.
27-32 Asks if directory is to be listed. Sets NF, reads directory and returns.
33 Requests name of required file.
35-68 Directory read routine. Flags an error (NF) if required file not found.
78-81 Reads file type entry (byte zero) and stores TP. The lower 4 bits are stored AF.
82-87 Checks file type and stores string in TY$.
88-89 Reads bit 7 of file type byte and stores in XX$ (File closed bit).
90-91 Reads bit 6 of file type byte and stores in SE$ (File secure bit).
92-97 Reads bytes 28 and 29 of file entry and stores in DK.
98-102 If relative file, length of record is read from byte 21 and stored in LE
103-116 Calculates the starting address in program file.
117-126 Calculates number of free blocks on disk.
127-128 Calculates side sector blocks of a relative file from LE and RF.
129-157 The results can now be printed either to the screen or printer.
158-164 Asks if another file is required.
165-168 Sub routines for switching screen off and on.

As it stands, the program is functional and stands alone. However, you may wish to include it as a sub-routine of a larger utility package. This can be achieved without too much trouble. You may try modifying the directory read section so that the directory is printed across the screen in two columns, instead of down the screen. Or, one which I like, is to put a four line window across the top of the screen so as not to spoil the screen layout.

PROGRAM ANALYSER

Programming can be made simple with these three C64 performance analysers -

COMMODORE 64 PERFORMANCE ANALYSER

Basic is a programming language which makes it very easy for programmers to create complex programs with a minimum of effort. We pay a price for this programming ease, and that price is often poor performance, that is, our Basic program runs slowly. Another problem which confronts the Basic programmer is what to do when a program runs without failing, but doesn't give you the results you expect. How do you find out what your program is doing without adding PRINT statements to your program to trace execution or interrupt execution at strategic points?

The PERFORMANCE ANALYSER helps to overcome both problems. Not only does it trace the logic flow in a Basic program, it also determines how long each Basic line took to execute. Thus the PERFORMANCE ANALYSER is a generalised performance analysis tool for the Commodore 64.

PERFORMANCE ANALYSER TRACE FACILITY

Most commercial traces usually amount to a window displaying five or six line numbers on the screen as your Basic program runs. The line numbers scroll in the window as each line is executed, and the window may or may not interfere with your program output. You normally cannot trace a Basic program which uses hires graphics, and you certainly cannot go back and check the line number sequence previously displayed. Although you can usually slow the trace display down (by the space bar for example), you have very little chance of writing down the line numbers on paper for a more detailed analysis.

The Performance Analyser overcomes all of these problems. It allows you to trace any Basic program which uses normal screen graphics, hi-res screens, sprites or sound and does not interfere with the operation of the program. The Analyser will not slow your program down, and allows you to give the trace display at your leisure. You may scroll backwards or forwards through the line numbers for as long as you wish.

PERFORMANCE ANALYSIS

The Analyser also provides you with a tool to determine how efficient your Basic program is. When it displays the line number it also displays the time it took to execute the line. As you scroll through the line numbers you can tell at a glance which line numbers are slowing down execution and which line numbers are executed most often Basic programs are the same as any other program - they follow the 80/20 rule. That is, 80 per cent of the work is usually done by 20 per cent of the program. The Performance Analyser is the tool you need to tell you which 20 per cent of your program is doing 80 per cent of the work, and how long it is taking to do it. You can then concentrate on making that part of your program more efficient.

ANALYSING A BASIC PROGRAM

The Analyser is written entirely in Machine Language, and is designed to cause a little interference as possible with the traced program. The Analyser is normally loaded at 38912, and all Analyser variables and constants are contained in the 2K from 38192 to 40959. Your Basic program thus has the RAM between 2048 and 38911, any low storage locations it requires and the free RAM at 49152. Should you require the RAM at 38912, then set the top of Basic pointer (55,56) to the last RAM location available to Basic, and the Analyser will use 2K of RAM before this address. For example, if the top of the basic pointer is set to 32768, then the Analyser will load itself at 30720.

Type in the Analyser loader program and save it as ANALYSER1. Make sure you verify that what you saved is correct. To use the Analyser simply issue a load "ANALYSER1" after setting the top of BASIC pointer if necessary. ANALYSER1 will set the required Basic pointers, POKE the Analyser Machine Language logic into the correct RAM, relocate all required ML addresses and print messages to indicate how to start and stop the Analyser and display the trace data. The following messages are displayed on the screen by ANALYSER1 during execution;

```
LOADING THE ANALYSER AT 38912
LOAD OK
RELOCATION OK
1 START ANALYSER = SYS 38912
2 STOP ANALYSER = SYS 38915
3 DISPLAY DATA = SYS 38918
```

If the load fails, or the relocation of addresses fails, a message is issued and ANALYSER1 stops.

Obviously to start the Performance Analyser you SYS to 38912 or to the address displayed by ANALYSER1. You can do this from a program or from direct mode. The message TRACE STARTED is displayed by the Analyser, unless you start it from a program. The message is not issued then to ensure that the Analyser does not interfere with program messages or displays.

After the Analyser has been loaded, you then LOAD the Basic program or programs you wish to analyse. The Analyser monitors execution of your program(s), and saves trace data in the trace data buffer for later display. If you only want to trace part of a Basic program, you would do the following;

```
1000 REM START THE ANALYSER
1010 SYS 38912
1020 FORTH = .6to6STEP.1
```

```
1030 X = A*COS(TH)
1040 Y = B*COS(TH)/C
1050 NEXT
1060 REM STOP THE ANALYSER
1070 SYS 38915
1080 REM DISPLAY TRACE DATA
1090 SYS 38918
1100 END
```

After your Basic program has finished, or you stop it executing, you can stop the Analyser if you want to. However, you don't stop it to display the trace data. You may leave it active to trace another program if you want to.

Obviously, to stop the Performance Analyser you SYS to 38915 or to the address displayed by ANALYSER1. You can do this from a program or from direct mode. The message TRACE STOPPED is displayed by the Analyser, unless you stop it from a program. Again the message is not issued to ensure that the Analyser does not interfere with program messages or displays.

```
-- PROGRAM STATISTICS --
PROGRAM NAME      = X-RAY FILES
PROGRAM SIZE      = 3916
NO OF LINES       = 169
NO OF COMMANDS    = 414
NO OF VARIABLES   = 25
 USE ANY KEY TO CONTINUE
```

Finally, you may display trace data at any time by entering SYS 38918 or SYS to the address displayed by ANALYSER1, and of course you may do this in direct mode or from a program. The message NO TRACE DATA is displayed by the Analyser if there is nothing to display. Again the message is not issued if you are under program control. This is to ensure that the Analyser does not interfere with program messages or displays.

If there is data to display the Analyser presents it in full-screen mode, that is a page or full screen data consisting of line numbers and line execution times is displayed and the Analyser ML program waits for you to press one of the function keys; F1 terminates the display, F5 scrolls back to the previous page of data and F7 scrolls forward to the next page of data.

You may scroll back and forward through the trace data for as long as you like with function keys F5 and F7. When the end of the trace data is found, the number of lines executed and the total execution time is displayed, and the Analyser ML program waits for you to press a function key. The Analyser will only recognise F1, F5 and F7 function keys. All other keys are ignored. If you scroll forward from the end of the display, you wrap around to the start of the trace data again. You can't scroll back from the top of the trace data, you may only scroll forward.

NOTE: Trace data will be displayed automatically when the trace data buffer area is full. The trace data buffer is actually the RAM under the BASIC ROM. As much trace data as possible is stored there before the execution of the Basic program is interrupted and the trace data displayed. If you want your Basic program to continue, simply press F1 and the trace display is terminated. Your program begins execution from where it was interrupted. If you want to browse the trace data, then use F5 or F7 to scroll back and forward through the data.

HOW THE PERFORMANCE ANALYSER WORKS

The Analyser works by monitoring the execution of Basic programs via the character dispatch vector in low storage. As each program byte is interpreted, the Analyser checks to determine if the current line number (57,58) has changed from the previous byte read. When the line number changes, then the Analyser stores the line number and current time in the trace data buffer under the BASIC ROM. This is done until such time as the trace data buffer is full.

When the buffer is full, the Analyser saves the first 2K of low storage (0-2047), colour RAM and various control registers in the RAM under the KERNAL ROM. The trace data is then displayed, and when the display is stopped via function key F1, the Analyser restores the first 2K of low storage, the colour RAM and the various control registers. This allows the Basic program to restart execution from the point were it was interrupted, and the program screen is restored, as well as character colours and backgrounds.

If your Basic program uses the RAM under the Basic of KERNAL ROMS, then you cannot analyse it with this utility. Note also that if your Basic program resets the time (TI$ = "000000"), then the Analyser will not fail, but the execution times displayed will be unpredictable.

COMMODORE 64 PROGRAM ANALYSIS

Commodore 64 Program Analysis (C64PANAL) is a Basic program which analyses the contents of any Basic program and displays the information on the screen or printer, C64PANAL first displays summary information which contains the program name, the size of the program in bytes, the number of lines in the program, the total number of lines in the program, the total number of commands (i.e. PRINTs, GOTO's, If's etc.) and the number of variables.

Once the summary data has been viewed, a detailed list of the commands used in the Basic program and the number of times each command is used is displayed. When you have finished viewing the variable data you may end the display, ask for the information to be re-shown or send the data to your printer.

USING C64PANAL

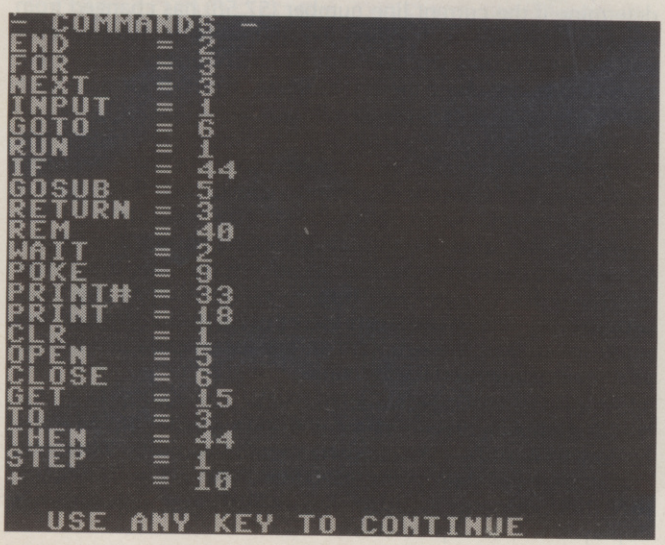
C64PANAL allows you to analyse your Basic program. It does this by running in the 4K of free RAM at 49152 to 53247, and loading the Basic programs it analyses at 2049. By not using the RAM between 2048 and 40960, C64PANAL is capable of analysing the largest Basic program. However, with only 4k of RAM to run in, C64PANAL will run slowly analysing large Basic programs because many garbage collections will be done to ensure that there is sufficient space for C64PANAL to operate correctly. Also, only 50 variables can be displayed because of space constraints.

Obviously if C64PANAL is to run in the RAM at 49152 then some changes need to be made to Basic addresses need to be

changed as well as the start of variables etc. These changes are handled by the C64PANAL loader program. LOADER is the Basic loader program which automatically loads C64PANAL. It sets the low storage pointers, and then uses the dynamic key facility to automatically load C64PANAL.

You must create and save LOADER first on tape or disk. Next type in C64PANAL and save it directly after LOADER on tape or on the same disk as C64PANAL.

Note that if you are using disk you need to change line no 10 in LOADER from LOAD "C64PANAL", 1,1 to LOAD"C64PANAL", 8,1 so that C64PANAL will be loaded from disk and not tape.



Once you have saved LOADER and C64PANAL to tape or disk then simply load LOADER and RUN it. LOADER will set the various low storage pointers and then set up the screen and keyboard buffer so that when it ends, C64PANAL is automatically loaded at 49152. When C64PANAL has been loaded it begins execution automatically, clears the screen and places the first message on the screen:

LOAD FROM DISK (Y/N)?

If you want C64PANAL to load the Basic program it analyses from disk, then reply Y. Otherwise reply N and the program will be loaded from tape. Before replying to this message, you should have the tape or disk which contains the program to be analysed in the datasette or disk drive.

The next message to be displayed:

PROGRAM TO BE LOADED = = ? Your answer to this message tells C64PANAL the name of the program it is to load from tape or disk to analyse.

C64PANAL then uses the KERNAL load subroutine to load the Basic program at 2049 and begins to analyse it. Since it may take some time to analyse large Basic programs, C64PANAL places the line number being analysed in the top left-hand corner of the screen while scanning the Basic program. When analysing is finished the summary report is displayed as follows:

——PROGRAM STATISTICS——
PROGRAM NAME = C64PANAL
PROGRAM SIZE = xxxxx
NO OF LINES = xxxxx
NO OF COMMANDS = xxxxx
NO OF VARIABLES = xxx

USE ANY KEY TO CONTINUE

You may view the summary report for as long as you wish. To move, to the command report, simply use any key and the following display appears on screen:

- COMMANDS-

END	= 1
FOR	= 5
NEXT	= 6
DATA	= 10
INPUT	= 1
READ	= 1
GOTO	= 25
IF	= 30
GOSUB	= 17
RETN	= 17
REM	= 8
POKE	= 5
PRINT	= 38
THEN	= 25
+	= 45
-	= 15
*	= 51
/	= 19
AND	= 1
=	= 55
MID$	= 3

USE ANY KEY TO CONTINUE

If all commands used in the program can be displayed on one screen, then when you press any key you will move to the VARIABLE display. If more commands are used than can be displayed on one screen, then the next screen of data will contain command data. When the last of the command data has been displayed and the USE ANY KEY message is displayed, when you press any key the list of variable appears on the screen. Note that + - * / = < and > are considered commands when used in statements such as A=A+B*C-D/E of IF X 0 GOTO 1000.

When the commands are finished the list of variables is displayed as shown

-VARIABLES-

I	= 2
X	= 4
RR$	= 5
Z	= 3
ZZ%	= 9

USE ANY KEY TO CONTINUE

When the last of the variables has been listed, C64PANAL displays the following message

R = RE-DISPLAY, X = END, P = PRINTER

If you press the R key, then all information beginning with the summary display is re-shown. If you press the X key then program execution is terminated and the final time message is displayed:

TIME TAKEN = xxxx.xx

This is the time in seconds it has taken C64PANAL to analyse your program. You may then use C64PANAL to analyse another Basic program.
Press P and the information is sent to the printer.

APPLYING C64PANAL

C64PANAL has many uses. You can find the size of your Basic program, the number of variables you use and the number of lines in your program. The number of lines is important because each line in a Basic program carries an overhead of 4 bytes (2 bytes for a link address and 2 bytes for the line number). A 500 line program uses 2,000 bytes of storage for link addresses and line numbers. If you have an excessive number of lines, you can conserve space by reducing the number of lines (also known as crunching your program). You reduce the number of lines by placing multiple commands on the same line separated by colons, removing blanks and removing REM commands.

Processing new lines also carries with it a performance penalty. The more lines in a Basic program, the longer it normally takes to run. By reducing the number of lines, you normally reduce produce execution time. C64PANAL will tell you how successful you have been at reducing the number of lines in your program. It will give the size of your program and the number of lines before crunching, and then after you have made your changes you can run it again and get the new figures.

The detailed list of commands (i.e. prints, GOTO's IF's etc.) can also be used to reduce program size and increase performance. For instance, if you find that you have a very large number of IF commands, then you may be able to reduce them by using the ON command. For example if you have:

IF CC = 1 GOTO 1000
IF CC = 2 GOTO 1100
IF CC = 3 GOTO 1200
IF CC = 4 GOTO 2000
IF CC = 5 GOTO 2100
IF CC = 6 GOTO 2200

then you could replace the IF commands with one:

ON CC GOTO 1000, 1100, 1200, 2000, 2100, 2000

It is also interesting to see the pattern of commands in various programs and which commands are used most frequently. In string operations the LEFT$, RIGHT$, MID$ etc. will figure prominently. However, the most common commands used are the IF, GOTO, FOR and NEXT and PRINT.

The list of variables is a powerful tool to help in the execution of your Basic program. Basic maintains a list of variables, and the closer a variable is to the start of that list, the less time that is needed to find the variable when it is referenced in a statement. For example, every IF X = 3.7 TOTO 100 is executed, the X variable must be found in Basic's list of variables' to check if it is 3.7 or not. Thus the closer X is to the top of the list, the faster it is found. The order of variables makes a significant difference to the execution time of your program if you have a large number of them. C64PANAL helps by giving you a guide as to which variables ought to be defined first so that they appear at the top of Basic's list of variables. You can ensure the order of variables by defining them in the following manner:

X = 0:A = 0:Y = 0:PC = 0:TK% = 0 etc.
X will come first, A second, Y third in the list and so on.

If you have Basic programs where execution time is crucial (for e.g. games programs) then C64PANAL will be an important tool to help you analyse those programs and make them faster.

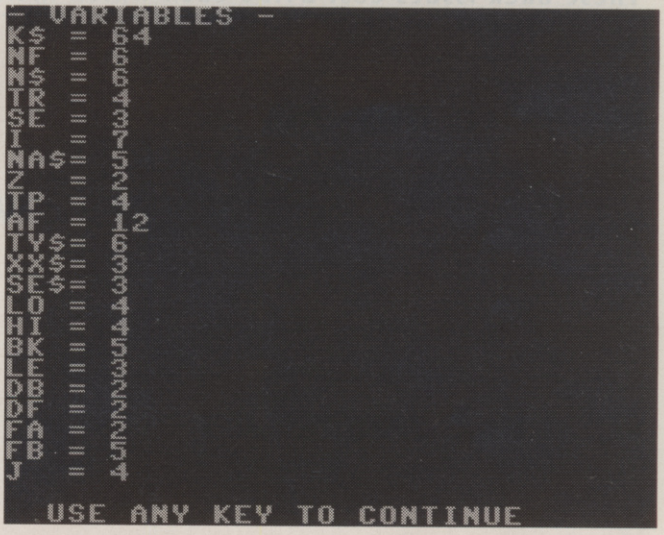
COMMODORE 64 SWITCH

Commodore 64 SWITCH is a short Machine Language (ML) program which resides in RAM just before the BASIC ROM. It occupies storage locations 40704 to 40959. C64 SWITCH allows you to partition your C64 into two logical machines. You switch between the two partitions or regions with a single key stroke. With this utility, you can load two Basic programs at once and compare them or work on them. However, you cannot have both programs running simultaneously.

USING C64 SWITCH

C64 SWITCH allows you to set variable region sizes. The regions are designated zero (0) and one (1) and region 0 will extend from location 2049 to the limit your set, while region 1 extends from the end of region 0 to 40703.

To use the switching function, simply load SWITCH, which is a Basic loader program. When you run it, SWITCH will load the ML routine at 40704 and display the message:



ENTER REGION 0 ENDING ADDR = = >?

You enter the ending address for region 0 (and thus region 1 starting addr) and the final messages are displayed:

REGIONS 0 and 1 INITIALISED REGION ACTIVATED = 0

To switch between the two regions use F1/F7 keys. F1 will activate region 0 while F7 will activate region1. The active region is displayed in the upper right-hand corner of the screen in reverse video. To deactivate the SWITCH, simply hit RUN STOP/RESTORE or turn the C64 off and on.

APPLYING SWITCH

C64 SWITCH has three main uses. You can load two Basic programs at once, and work on them or compare them. You can use region 1 as a data region which is accessed by a program in region 0 (SWITCH was originally written for this purpose). Finally, you can use SWITCH as a means of merging two programs. If you want to add code to a program in region 0 from a program in region 1, simply LIST the statements in region 1 on the screen, press F1 to activate region 0 and then move the cursor over the lines you want added and press RETURN. Each line will be entered into the program in region 0.

NOW IS THE TIME TO CATCH UP ON ISSUES YOU HAVE MISSED

The following back issues of CDU are still available direct from ALPHAVITE PUBLICATIONS LTD. Please note that if ordering one of the following back issues, you will receive a copy of the disk, along with photostat copies of instructions for the relevant disk programs ONLY. These back issues cost £4.50 each which includes Post/Packing. Please make cheques/Postal Orders out to:- ALPHAVITE PUBLICATIONS LTD (Allow 28 days for delivery).

VOL 1 No.1 NOV/DEC '87

DIRECTORY DESIGNER - Tidy up your disks with this Editor/Designer.
TEXT ENHANCER - Improve your text displays.
MOBSTER - Have you got what it takes to be a gangster.
3 INTO 1 PLUS - A superb Character, Sprite and Background Editor.
SKI RUN - All the thrills of the slopes with this game.
SPRITE PRINTER - Dump your favourite sprites onto your CBM printer.

VOL 1 No.2 JAN/FEB '88

DISK LIBRARIAN - Keep track of what's on what disk.
DISK MATE - Handy pop-up disk functions.
NOLUXE PAINT - A superb low-res drawing package.
TEXT CRACKER - Grab those character sets you like for your own use.
QUAD - New life for the brick/bat game.
FIVE-UP - Can you win at this dice game?
RAM DISK C128 - Our first program for the C128.

VOL 1 No.3 MAR/APR '88

SUPER-TACT - Tactics are the essence of this game.
CHAOS IN SPACE - A shoot-em-up that's deceptively different.
C-ZAP - Speed is the name of the game with this compiler.
BASIC+ - A comprehensive Basic extension.
TAPE ARCHIVE - Be safe and back up your disks.
LINK & CRUNCH - Running out of memory/disk space? Not anymore.
PSYMON - A full-facility machine code monitor.
DISK LIBRARIAN II - An updated version of DL.
C128 AUTOBOOT - For C128 owners - load C64 progs at C128 speed.

VOL 1 No.4 MAY/JUN '88

DRUMSYNTH - Percussive programming.
C128 PULL DOWN WINDOWS - Windowing for the C128.
TOKENISER - Word-process your Basic programs.
C-CAD - Enter the world of Computer-Aided-Design.
BASIC COMPACTOR - Squeeze up your Basic programs.
SANTOLUS - A demanding smooth-scrolling maze.
ATLANTIS - Explore the lost continent.

VOL 1 No.5 JUL/AUG '88

DISK TOOLKIT - The Editors very own comprehensive utility for disk users.
RELOCATOR - How to move your machine code.
MIND GAMES - Unscramble the Presidents brain.
3-D BREAKOUT - Bash those bricks in 3 dimensions.
PEGGY 128 - An amusement for C128 owners.
ORRERY - Planetary positions computed.
MESSAGE CONSTRUCTION KIT - Full-screen scrolling messages.

VOL 1 No.6 SEP/OCT '88

SCORPION - If it moves, kill it.
COLOUR MATCH - Tailor your 64 screen colours to your own taste.
C128 SPREADSHEET - Accounts can be simple.
ESCAPE - Can you find a way to escape the Nazis?
STARBURST - Your chance to save the galaxy.
SCORE KEEPER - Using Sprites for your game scores.
ADDIT - A tactical numbers game.
LOCATION FINDER - Find out what that bit of code's up to.
FRACTAL FROLICS - Fun with the Mandelbrot set.

VOL 2 No.1 NOV/DEC '88

CDU FORTH - Escape from Basic with our compiler.
TEXTED - Word-processing made easy.
EXTRACTOR - Build up your sprite library.
WINDOWS 64 - Generate windows painlessly.
ZMON - Program your C28's Z80 chip.
CRIBBAGE MASTER - A C64 first, this program plays a mean game.
OBLIVION! - Fight off the deadly Janoids.

VOL 2 NO.2 JAN/FEB '89

DISK TURBO - Speed up your disk access.
BLASTBALL - A bat 'n ball extravaganza.
COLOUR BIND - A sliding block problem with a difference.
BORDER SPRITE - Make use of those screen edges.

DATA MAKER - The easy way to get machine code into Basic.
LIFE - The bizarre world of cellular automata.
MENU MAKER 128 - Make your 128 disks easy to use.
MICRODOT - Save the world from radiation sickness. (Superb game)
SPOTS - Can you beat the machine's dice rolls?
EASY SCROLLER - Scrolling made simple.
RUNAWAY - Can you escape from a dreary domestic existence?
LOGIC - The puzzle that will have you tearing your hair out.

VOL 2 No.3 MAR/APR '89

DARTS - Pub fun - in the comfort of your own home.
CDU PAINT - The ultimate C64 graphics package.
DEVAID - The Editors very own extended Basic with 41 new commands.
BAZAIR - Can you survive the maze of death?
ARAKNIFOE - Get your own back on those creepy-crawlies.
C128 GRAPHICS PRIMER - Make use of those 80 columns.
DOMINOES - Pit your wits against Max and Joe.
PHANTOM - Strike a blow for world revolution.

VOL 2 No.4 MAY/JUN '89

BASE ED - Get organised with this C64 database.
DBASE 128 - 40 or 80 column storage for C128 owners.
6510+ - The ultimate in C64 assemblers.
SID SEQUENCER - Make commodore music with ease.
LIBERTE - Escape the POW camp in this 1940's style adventure.
FX KIT - Bangs, Pows and Zaps made easy.

VOL 2 No.5 JUL/AUG '89

FONT FACTORY - Create your own characters.
HIRES DEMO KIT - Add music to your favourite picture.
ANIMATOR - Get those sprites moving
BORDER MESSAGE SCROLL - Say what you like along the bottom of the screen.
TYPIT-128 - Create professional text layout on your C128
SCREEN COPIES UTILITY - Download your favourite screens, including CDU paint files.
VIDI-BASIC - Graphic based extension to Basic.
64 NEWS DESK - Become a C64 reporter.

VOL 2 No.6 SEP/OCT '89

MICKMON - An extensive M/C monitor.
SCRAPBOOK - Collectors and hobbyists database.
CELLRATOR - Enter the caves if you dare.
RAINBOW CHASER - Rainbows means points in this unusual game.
HIDDEN GRAPHICS - Utilise those graphic screens.
FORTRESS - Save the world! Yet again.
DISK HUNTER - Keep tabs on your disk library.
SUPERFILE - One more for the record keepers.

VOL 3 No.1 NOV '89

BASIC EXTENSION - Windows and Icons the easy way.
B-RAID - Vertical scrolling shoot'em up.
DISKONOMISER - Prudent disk block saving.
HELP - Design your own information help screens.
ORSITAL - An arcade style game with a difference.
PROGRAM COMPARE - Basic program development has never been easier.
RASTER ROUTINES - A few colourful demos.
SPRITE EDITOR 1 - A no nonesense basic sprite editor.
WABBIT - Help the rabbit collect his carrots.

VOL 3 No.2 DEC '89

KRON - Can you meet the challenge?
CDU MENU KIT - Design your own menus with ease.
PHOBOS - Break out of jail and gain your freedom.
LIMBO - Collect the cells off the blocks.
MUSIBASIC - Sound and music made easy.
PANIC - Is your eye as quick as your joystick.
TEMPLATE DESIGN - Backgrounds the easy way.
QUIKWORD - You very own expandable word processor.

VOL 3 No.3 JAN '90

4 IN A ROW - Connect a row of counter.
FROGS IN SPACE - Leap to safety across the space lanes.
BLACKJACK - Don't loose your shirt.
LORD OF DARKNESS - Defeat the evil lord in true adventure style.
MARGO - Fly around and collect jewels and fuel.
JETRACE 2000 - Have you got what it takes to be best?
ULTIMATE FONT EDITOR - Create your own screens, layouts and characters.
SELECTIVE COLOUR RESTORER - Design your own system start up colours.
6510+ UNASSEMBLER - Transform 6510+ M/C into source with labels.
TRIVIA CHALLENGE - The first of 3 files for this superb game.

VOL 3 No.4 FEB '90

COLOUR PICTURE PRINT - Download your favourite colour screens.
BASE-ED2 - An update to our popular database system.
1ST MILLION - Play the market in this strategy game.
FM-DOS - Enhance your drives operating system.
GEOS FONTS - A further 4 fonts for Geos users.
HASHING IT - Reletive file programming made easy.
MULTI-SPRITE - Make full use of up to 24 sprites.
DIRECTORIES EXPLAINED - Find your way round the directory jungle.
TRIVIA CHALLENGE - The 2nd part.

VOL 3 No.5 MAR '90

PLAGUE - Become your planets Guardian and Defender.
SURROUND - Reversi on the C64
GEOS FONTS - The last of 12 new Geos fonts.
SCREEN SLIDE - Create your own slideshows.
JOYSTICK TESTER - Put your stick(s) through the mill.
COLOUR MATCHER - Mastermind for the younger players.
SCREEN MANIPULATOR - Full use of the screen now obtainable.
VIDEO RECORD PLANNER - Keep tab on your home recordings.
TRIVIA CHALLENGE - The 3rd and final part of the game.

VOL 3 No.6 APR '90

BAR PROMPTS - M/C input routine.
HI-LITE BARS - Input routine but in Basic.
TEXAS DEMO - Example of using Basic in demos.
CHARS TO SPRITES - Convert UDG's to sprites.
FONT FACTORY - Complimentary program to the above.
3D-TEXT MACHINE - Impressive 3D text screens the easy way.
SCREEN ENHANCER - Makes full use of the screen easy to achieve.
SPREADSHEET 64 - An excellent, easy to use spreadsheet.
MINI-AID - 3 short utilities to aid the Basic programmer.
C128 COLLECTION - 3 very useful C128 programs.

The following back issues can be obtained from Select Subscriptions Ltd, River Park Estate, Berkhamsted, Herts, HP4 1HL. These back issues cost £3.75 each which includes Post/packing. Please make cheques/postal orders out to:- SELECT SUBSCRIPTIONS LTD (Allow 28 days for delivery).

VOL 3 No.7 MAY '90

NUDGE - FLD explained in laymans terms.
WINDOW WIPER - An alternative screen wipe system.
CHARACTER EXTRACTOR - Borrow those nice character sets you see.
MAZE GENERATOR - Create your own fun.
HIRES ANIMATOR - This difficult subject made easier.
SPRITE DRIVER - Platform game designing without the fuss.
ROTOTRON - Demonstration of Sprites and Sound.
TEXT COMPRESSION - How to squeeze a gallon into a pint.
SCREENS - Make up your own help screens and keep them in memory.
INTERRUPT POINTERS - Geos style windows and pointers for you.

VOL 3 No.8 JUN '90

ALEATORY MUSIC - An alternative music system.
SPRITE BASIC - Efficient sprite handling through Basic.
SPRITE GENERATOR - Another sprite editor for your library.
MUNCHER - Pacman returns with a vengeance.
ASTRODUS - Escape the spaceship Astrodus in this adventure.
1581 DIRECT ACCESS - Find your way around the 1581 disk drive.
PERSONAL ORGANISER - Design your own organiser pages.
128 CONVERTOR and MATHS AID - 2 more for C128 users.

VOL 3 No.9 JUL '90

QUICK MERGE 64/128 - Another useful routine for your archives.
THE GAME PLAN - An aid to knowing what where in your games.
CHARACTER dESIGNER - Another designer for those without.
HASHBASE 128 - A powerful database for C128 users.
REVASM 64/128 - Two unassemblers for non Speedy Assembler owners.
SPEEDY UNASSEMBLER - An unassembler specific to Speedy Assembler users.
BANKS AND MEMORY - An aid to redifining screen and graphic memory.
GRAPHICS FACTORY - A novel way of getting in graphic design.
POT POURRI - A selection of useful routines for all users.

VOL 3 No.12 OCTOBER '91

ROLL'EM - An example of using Graphics Factory.
COLOUR MATCH - A short utility for C128 users.
SPREAD-ED - The third in the 'ED series.
RASTER EDITOR - Put those raster lessons to good use.
ADDRESS BOOK - A somewhat unusual address base.
SUPERSORT 64/128 - Sorts have never been easier.
SPRITE EDITOR C128 - Another utility for C128 users.
GRAPH-ED - The last in the 'ED series.
BACKGAMMON - That popular board game gets an airing.

VOL 4 No.4 FEBRUARY '91

CRANMORE DIAMOND - Another Tony Rome adventure.
COMPACTOR - Program protection made easy.
ADVISOR - Artificial intelligence on the 64.
GALACTIC ENCOUNTER - Battleships played out in space.
IRQ64' - Interrupts on the C64 explained.
2 FOR THE C128 - Check your 128's RAM and PLAY command.
CHEQUE BOOK ORGANISER - Organise your cheque books and statements.

VOL 3 No.10 AUG '90

LIMBO 2 - The sequel to Limbo
SCREEN DESIGNER 128 - Screen designing made easy.
DATABASE78 - A database full of features.
LETTER MAKER - Text Screens made decidedly pleasing.
FUNCTIONS - Make full use of those function keys.
GAMES LIST CREATOR - Keep tabs on your games disks.
DUAL DISKCOPY - At last an intelligent disk copy program.
SEQUENCER64 - Musicians have a field day.
SECURITY - Put all those broken joysticks to good use.
SUPERBOOT! - Auto load your programs.

VOL 3 No.11 SEP '90

BANKING 128 - A simple way of keeping your money straight.
DISK DRIVER V4 - A simple disk utility.
AUTOBOOT 128 - C128 users get easy access to CDU programs.
READING BETWEEN THE LINES - Build your own adventure parser.
I.D.O.S. - A comprehensive drive utility.
PRICE CALCULATOR - Keep tabs on inflation.
B.O.S.S. - Yet another alternative to the standard Basic.
SCREEN DESIGNER/COMPILER - Impressive screen layouts for all.
LANDSCAPE ROUTINE - Beginners guide to scrolling backdrops.
SAMPLE KIT 64 - More sampling for all you musicians.

VOL 4 No.1 NOV '90

WAR AT SEA - C128 version of battleships.
FULL DISK JACKET - Keep track of your disks.
NEAGOX - Blast everything that moves.
NUMDEF - A basic game to test the reflexes.
MEMORY SCANNER - Look through memory the easy way.
MONEY 64 - Budget planning for the 90's.
XINOUT - An alternative input routine.
CALENDAR C128 - No more buying of calendars.
GOMOKU - An interesting variation of 'GO'.
SMOOTH SCROLL DEMO - Create your own scrolls.

VOL 4 No.2 DEC '90

ILS (German Program) - A C128 language tutorial.
SCREEN DESIGN CORRECTION - An update to this excellent utility.
BETTER BACKUPS - Help for ARC users.
MACHINE CODE GEMS - A suite of MC routines to aid you.
COLOUR TABLE EDITOR - Get those raster bars right.
MULTITASKING C128 - An implementation for the C128.

VOL 4 No.3 JAN '91

2X2 CHARACTER EDITOR - Design large fonts easily.
ENCRYPTION - Password and program protection.
SECURE - Another against would be hackers.
KA43/5 OPEN SYSTEM - Expand the C64's op system.
32 SPRITES - The world of sprites opened up.
DISPLAY.ASM - Multitasking source file displayer.
TERMINUS - A demo that is somewhat unusual.

Lite Menus

You can move the cursor to any position on the screen by declaring 2 variables and calling the subroutine, as follows :-
X - X position of the cursor (0-39).
Y - Y position of the cursor (0-24).

Once these variables have been set up, you call the routine by GOSUB 12000.
Example of PRINT AT.

```
X=5:Y=7:GOSUB 12000
```

This will move the cursor to (5,7).
So, using all three techniques can make your menus quite impressive.

How to add LITE MENUS! to your own programs.
There are 2 ways to add LITE MENUS! to existing programs.
1. If you have an extended basic utility, use its APPEND function.

2. Or use this.

```
LOAD"<your program>",8
POKE 43,PEEK(45)-2 : POKE 44,PEEK(46)
LOAD"LITE MENUS! O/S",8
POKE 43,1:POKE 44,8
```

You should find now that LITE MENUS! will have been added to the end of the program.

TROUBLESHOOTING.

You will have to add 2 to the longest string to fit inside the box. EG :-
QUIT is 4 characters so X1=6

Also, you have to add 2 to the number of items in the array, EG :-
DIM O$(5), so Y1 for the box = 7

Also, try not to mix up X and Y co-ords.
Notes.

Control of the hi-light bar is done by <CRSR UP> (move up), <CRSR DOWN> (move down and <RETURN> for select.
I have noticed that the shorter the list, the flash rate of the hi-light bar is faster, I have not worked out why yet?!

The SYS call is to $FFF0 (decimal 65520) which is the (X,Y) position set.

The present routine cannot be compiled by itself, but once in a completed program, it runs very smoothly when compiled. I hope that this program will help somebody out there, and if you do use LITE MENUS! in your own programs, an acknowledgement would be nice.
If you need to see **LITE MENUS!** in action, load the demo by typing **:-LOAD"LITE DEMO",8**

STRATEGY ADVENTURE

C64 disks only

INFOCOM		HILLS FAR	£19.95
BALLYHOO	£14.95	OVERRUN	£24.95
BUREAU CRAZY C128	£14.95	PANZER STRIKE	£24.95
HITCHIKERS GUIDE	£9.95	PHANTASIE III	£24.95
LEATHER GODDESS	£9.95	POOL OF RADIANCE	£24.95
		PRESIDENT ELECT	£14.95
INTERSTEL		QUESTRON I	£19.95
EMPIRE	£29.95	QUESTRON II	£19.95
		ROADWAR EUROPA	£19.95
LUCASFILM		SECRET OF SILVERBLADES	£24.95
ZAK MC KRACKEN	£14.95	SIX-GUN SHOOTOUT	£12.95
		STORM ACROSS EUROPE	£24.95
MICROLEAGUE		TYPHOON OF STEEL	£24.95
MICROLEAGUE BASEBALL	£24.95	WAR OF THE LANCE	£24.95
MICROLEAGUE FOOTBALL	£24.95	WARGAME CONSTR SET	£19.95
MICROLEAGUE WRESTLING	£19.95		
		SUBLOGIC	
SSI		FLIGHT SIMULATOR II	£24.95
50 MISSION CRUSH	£12.95	NIGHTMISSION PINBALL	£12.95
BATTLES OF NAPOLEON	£24.95	STEALTH MISSION	£24.95
BUCK ROGERS	£24.95		
CHAMPIONS OF KRYNN	£24.95	**TELARIUM**	
CURSE OF AZURE BONDS	£24.95	DRAGON WORLD	£9.95
DRAGON STRIKE	£24.95		
FORTRESS	£12.95	**WIZARD**	
GEOPOLITIQUE 1990	£12.95	SUPERSTAR ICE HOCKEY	£14.95

Hint Books £7.95 Each:
BARDS TALE I, II or III, BUCK ROGERS, CHAMPIONS OF KRYNN, CHAOS STRIKES BACK, CURSE OF AZURE BONDS, DRAGON WARS, DRAKKHEN, DUNGEON MASTER, ELITE, ELVIRA, INDIANA JONES L.C. ADV, MANIAC MANSION, MIGHT & MAGIC I OR II, NEUROMANCER, POOL OF RADIANCE, SECRET OF SILVER BLADES, STAR FLIGHT, ULTIMA III, IV, V or VI, WASTELAND, ZAK MC KRACKEN.

Mail order only. Please allow 28 days for delivery.
Please make cheques and postal orders payable to CINTRONICS LTD.
Free post and packaging within the UK. Europe add £2 per item. Overseas add £4 per item.

CINTRONICS LTD.
16 Connaught Street,
London W2 .2AG

TIMEWORKS SOFTWARE

C64 GAMES ON DISK

Warriors of Ras	The Standing Stones	Zork 1
Web Dimension	Hopeless	Zork 2
Master fo the Lamps	Deactivators	Zork 3
Beamrider	Ankh	Supended
	Sky Runners	Starcross
Hero	Bugsy	Deadline
Pitfall	Cyborg	The Inheritance
Toy Bizarre	Chameleon	Valhalla
Pitfall 2	Spindizzy	Tarzan
Zenji	Dandy	Deactivators
The Tracer Santion	Mision Elevator	The Vikings
Pastfinder	Leviathan	Super Zaxxon
Rock'n Bolt		Beach Head
Escape from Paradise	Prodigy	Idris OC
Saucer Attack	Druid	Brian Jacks
Bug Blitz	Empire	Challenge
Wizard	Alleykat	Ace
Wild West	Idris OC	

BOOKS FOR THE C64 AND 128

C64 Machine Language	£6.95	C64 Tricks & Tips	£8.95
C63 Peeks & Pokes	£3.95	Idears BookC64	£3.95
Cassette Book C64	£3.95	Tricks & Tips C128	£9.95
Maths Tutor for C64	£7.95	Micro Wars On C64	£5.95
Power Plays on C64	£6.95	Ofical C64 Ref. Guide	£14.95

PRICE £5. each or any 6 for £25. Please give alternatives if possible since stocks are limited. Post free UK or overseas.
Access & visa acceped.

DTBS
18 NORWICH AVENUE, ROCHDALE LANCS OL11 5JZ
Tel/Fax: 0706-524304.

HACKMAN & Sobbin 2

Jenni Simpsons' continuing tale of woe and abandonment. Computer widows take note!!

The protruding, pulsating vein upon his furrowed forehead throbbed rhythmically in time to the beating of his clenched fist on the hallway wall.
"Tut, bloomin' hell", he exploded with fury, the expression within his eyes taking on an almost demonic appearance. "If I don't give Jamie an answer soon about that Amiga, I know I'm going to lose it. He'll sell it to somebody else, I just know he will", he wailed in agony.
"It's a lot of money", I said concernedly, and then, upon witnessing the devastating effect of my casual remark, added with exaggerated fervour, "but you are expecting a gloriously handsome cheque soon from that good and wonderful Archangel Eves, so we can easily afford it, can't we?

Now, as in one of Walt D's good 'ole heartwarming cartoons, the sombre mood immediately lifted, and the atmosphere suddenly and miraculously became charged with a magical electricity that literally bathed the now cheery reality in a warm and fairyland, technicolour glow. Flashing me an 'Errol Flynn' grin, the noonday sun glinting off a side incisor to dazzle and almost blind me with it's sheer brilliance, 'El Bono, as rapidly as one could utter 'Doc Jekyll and Mr. Hyde, had laced up his turquoise and grey Reebok's, slung on his old zippy jacket, and was half way down the garden path.

"Be back in a mo Doggy", he informed me with a cheeky grin, "just going to bell Jamie to tell him THE AMIGA IS MINE". And do you know what, just at that precise moment, I could have sworn I heard a deep and resonant echo. At power crazed voice exploding from a fanatical and obsessed being. "MINE, ALL MINE", boomed the threatening, manic words, followed by a cacophonious eruption of warped and cackling laughter, that seemed to evade the still evening air with it's neurotic promises of 'profound obsession', which were as apparent as a skinhead 'National Front' member, attending peacefully at a 'black is beautiful rally'.

The big day finally arrived, and 'El Bono, as excited as a 'Rottweiler' let loose in an orphanage, set off to catch the 11.38 train to Edinburgh. He was due to meet Jamie (the Amiga's owner), at our good buddy Mally's, at 12.30 pm, in order to view the Amigo (Ooops sorry, that's a Mexican Bandido, isn't it?). I meant of course the 'Amiga'. "See you later", I called as I watched him hurrying away looking remarkably like an absent minded stick insect. "Yep, see you at Mal's", he answered with a grin, whilst he frantically checked his pockets for various odds and ends.

After doing a little window shopping in the big and speedy city, I arrived at Mally's during the late afternoon. "Has it arrived?" I asked him, in a cheery tone as he opened the door to me.
"It certainly has," he replied, proffering me a resigned look that if it could have spoken would have said, 'You've really lost him now, you know!'

Upon entering Mal's living room, or should I say the flight deck of 'Concorde', for such an array of dials, screens, buttons, knobs and literally trows upon rows of 'gadgetobelia' the like of which are rarely seen, I was absolutely convinced that even the 'Starship Enterprise' itself, would have resembled a tatty cardboard cut out from the back of an old Corn Flake packet, in comparison to Mally's 'boudoir of scientific thingys'. And speaking of 'Startrek', seated at the helm of the 'Amigo' (Whoops, there goes that Mexican Bandido again - quick, someone send for the 'man with no personality, and I 'do' mean John Majors!) sat the very man himself, Captain Kirk, or more appropriately in this instance, Captain Smirk, for the size of his grin, could, I feel sure, have stretched from 'here to eternity', (and we all know of a 'blunderfool' movie by that name, don't we children?)

"Hi there", I announced in my best and cheery 'I've just arrived voice', patting John lovingly upon his hunched shoulders.
"Hi there", came back the strained and strangled 'oh please don't disturb me now' tone, that we 'breezy and what's happening then' folks, have all come to know so very well! Trying desperately to tear his gaze away

from the 'hypnotic and come hither' gleam of his 'new toy', he managed, after several attempts, and only minor whiplash this time, to turn his body, along with an extremely reluctant cranium, to greet me.
"What do you think?", he oozed with extreme joy, the magnetism of the shiny new monitor, managing to overcome his willpower completely, as he made one last feeble attempt to try to continue to hold my gaze.

B..O...I...N...G., suddenly his features had disappeared. Just like stretching a short length of elastic to full capacity, his body had snapped back into position, and he was once more, lost within that deep and mind-warping labyrinth of the uncharted universe of computing.

Mally shrugged his shoulders, shook his head gravely, and then in way of expressing his deepest sympathy, offered to make me a mug of his finest coffee.

"MALLY!". The desperate plea of one who is without doubt most definitely up 'faeces creek' without anything that even vaguely resembles a paddle, abruptly ruptured the peaceful silence like the anguished utterances of a small child who, upon suddenly discovering that after waiting an age in the classroom with their arm thrust high into the air in order to ask the teacher if they may be excused, now finds, that there is definitely no need. Ooops, yep, you've guessed it....a pool! A great, gaping pool of mass confusion, whose turbulent ebbings of frantic indecision spread throughout the room, to saturate each and every crevice with it's lethargic and ponderous viscosity.

The kindly offer of coffee, now as forthcoming as Ken Dodd's keeness to reveal his tax information to the Inland Revenue, Mally and 'El Bono promptly disappeared. I use the term 'disappeared' in this instance because just as a male dog becomes completely besotted with his female counterpart when her season is due, so to do computer freaks (Tut tut... I do apologise, er.....Computer Enthusiasts) become utterly locked (if you'll excuse the fun, I mean pun) into whatever it is that Computer peeps do. For short of a hand grenade, or indeed a large, icy cold bucket of water, nothing seems to be able to prise them apart from the beloved objects of their individual affections, be they cuddly and hairy, or indeed sleek and sophisticated.

However, now entirely beverageless, and in addition grossly unloved, the poor lonesome heroine of this sordid, miserable tale made her sad, sorry way slowly across the wide, desert like expanse of the honey beige carpet to resignedly create for herself, and may I add, her vanished duo of heartless tormentors, three magnificent cups of finest Nescafe.

The steaming, khaki liquid, now ready for consumption, I padded my way towards the dynamic duo and awaited my opportunity.....

"It's so simple John," Mally was saying, with extreme exasperation, "Tch, you've just go to get into the file that you want!".
"But how do I do that?" whined John, tears of sheer bewilderment filling his strained and reddened eyes.
"Well, you've got to enter the direction and then follow the path it's in," replied Mally, in a tone which implied that even the likes of Frank Bruno would be able to comprehend such a simple procedure. Know wot I mean 'Arry!
He hammered loudly and dramatically upon the keyboard for a few moments, and then cursed. "Blasted Amiga! Why can't they use standard MSDOS! What does D F Oh, mean?"

Daft Fool Obviously, thought I.

"On a PC," moaned Mally in continuation, "you just type A or B. But here it's DF1 or DF0"
"Yeah," groaned John in agreement. "On the ole' 64 it was so simple, I just had to type in the program name, hit the return key, and it was loaded!"...

I couldn't believe my luck... A minuscule interlude... Here was my big chance. "Er Mal, John, coffee," I announced, with some hesitancy whilst carefully placing the boldly emblazoned mugs which stated most emphatically, that they were, without doubt, and I quote: Allergic to Morning (Wow, I didn't realize that pottery could be so sensitive), within a relatively clutter free zone upon the mat black computer desk.

Save for Mally tappity-tapping upon his trusty keyboard, and two pairs of eyes glued to one shiny monitor, the moments passed silently by. Would my diligent and saint like activity again go completely unnoticed and unappreciated? Deep in my heart of hearts I was all too afraid that it would. Bitter past experience had taught me not to expect too much, for trying to get a teensy weensy reply or even a mini grunt of acknowledgment from a 'possessed keyboard fanatic', was rather like trying to extract an admission of guilt from our ex-prime minister. Virtually impossible.

'So there you have it', as our goodly film buff Barry Normal himself would say. That was the situation, and I know that you can all only too well imagine how I was feeling. Yes indeedy, not unlike our goodly, Geordie, geezer Gazza must have felt on that dismal day during the scintillating summer of '90, yep, you've guessed it, utterly downhearted and completely rejected. Cue, subdued lighting and heavyhearted music... Haunting overtones of stradivarius violins played by a sombre group of individuals, not like our good ole' Nige', who are without doubt suffering from terminal depression.

'Aaaaah...' I hear you all cry in pity. 'That poor little doggy, she has been so ill-treated. Please, let me take her home immediately, where she will be absolutely free from all 'computer paraphernalia', and promptly cosseted to death!'

To be continued.....

Commodore DISK·USER

Lineage: 58p per word. (+VAT)
Semi display: £11.50 plus VAT per single column centimetre minimum 2cm. Ring for information on series bookings/discounts.

All advertisements in this section must be prepaid.
Advertisements are accepted subject to the terms and conditions printed on the advertisement rate card (available on request).

0908 569819

Send your requirements to:
CDU CLASSIFIED DEPARTMENT, ALPHAVITE PUBLICATIONS LTD.,
20 POTTERS LANE, KILN FARM, MILTON KEYNES, MK11 3HF.

PUBLIC DOMAIN SOFTWARE

for the C64/128.
We have 150 disks full of all types of programs
All disks are £2.95. each

NOW FOR THE+4
From £3.75 per Disk.
Send SAE or phone for free catalogue (state model)

Kingsway Computers Services
72 Glencoe Road,
Sheffield S2 2SR
Tel:
0742 750623 / 0742 780222

HOME WORKERS WANTED
.(Mailing Enverlopes) £3 per Hr.
STAMPED S.A.E. TO
MAT KIZA
27 WOODSIDE PLACE
GLASGOW G37QL.

REPAIRS

C64, C+4.....£30. inc.
C128. 1541.....£40. inc.
Three month warranty, 1 week turnrounnd

SPARES

906114.... £9.30
6510.....£11.50
6526.....£11.95
6581.....£14.95
6569.....£21.50
8501.....£11.75
C64 Power Pack £22.00
C64 Cassette unit £22.00

OASIS COMPUTER SERVICES
Dept 4C. 14 Ridgeway Road,
Salisbury, Wiltshire, SP1 3BU
Tel:(0722) 335061

COMMODORE 64/128 SOFTWARE LIBRARY

☆Life Membership
☆Updates
☆Teenage Mutant Hero Turtle Tape £1.75
☆ 7 Day Hire
☆Tapes From £1.50 to £1.75
Also Disks.

FOR FREE CATALOGUE S.A.E TO -
ACE SOFTWARE LIBRARY,
14 CHIPENDALE COURT, BELFAST, BT10 0DU

ONLY POOLS AND HORSES

Every program written by a mathematician who has spent many years in the betting industry. Programs that utilise the tried and trusted methods of the professional, not pie in the sky theories that fail to pass the test of time.

FOOTBALL BOXFORM Written by a former pools expert for Littlewoods. The program has forecast over 50% more draws than would be expected by chance.
Homes, Aways and draws shown in order of merit and true odds given for every match. Merit tables show at a glance the teams currently in form and those currently having a lean spell. Australian pools program included in the price.

POOLS PLANNER by the same author. Full details given of 369 easily entered block perms ranging from 9 to 73960 lines and from 12 to 56 selections. All are accepted by the pools firms and are checked in seconds by your computer.

RACING BOXFORM Course characteristics (built in to the program) as well as the form of the horses are considered to speedily produce an order of merit for each race. Designed for flexibility allowing users to amend the program if they wish. **Price** still includes the highly acclaimed HANDI CAP WINNER - more than 1000 winners every year - over 25% of them at 5/1 or better. Order two or more and receive **FREE** a program
to work out almost any bet. So good its used by bookies,

Prices (Tape) £15.95 each. £25.95 any two. £35.95 all three. For discs please add £2. per program.

Advertised for Six years in the sporting and computing press.
BOXoft CLEVER.. GET THE BEST

BOXOFT (CDU), 65 Allans Meadow,
Neston, South Wirral L64 9SQ
Chque/ P.O./Access/Visa Tel:051-336-2668

VISA

BBC B & MASTER, AMSTRAD CPC & PCW COMMODORE 64/128, SPECTRUMS.

CLASSIFIED COUPON

ALL CLASSIFIED ADVERTISEMENTS MUST BE PRE-PAID.
THERE ARE NO REIMBURSEMENTS FOR CANCELLATIONS.

CDU CLASSIFIED DEPARTMENT, ALPHAVITE PUBLICATIONS,
20 POTTERS LANE, KILN FARM, MILTON KEYNES, MK11 3HF.

RATES:Lineage 58p per word (+VAT). Semi-display: £11.50 (+VAT) per single column cm minimum size 2cm. Series discounts available.

I enclose my Cheque/Postal Order for £.....for .....insertions, made payable to Alphavite Publications.
(Delete as necessary)

PLEASE DEBIT MY ACCESS/VISA CARD NO.:

EXP. DATE:.....

£.....FOR.....INSERTIONS

Name:.....

Address:.....

.....Post Code: .....

Daytime Tel No.: .....

Signature: .....Date:.....

❑ FOR SALE ❑ SOFTWARE ❑ SPECIAL OFFERS ❑ OTHER

TO ADVERTISE ON THIS PAGE OR OTHERS TEL: 0908 569819

...it's dynamite!

POWER CARTRIDGE

FOR YOUR COMMODORE 64/128

"...unbelievable value for money" ZZAPP! Dec 89

- POWER TOOLKIT
- POWER MONITOR
- TAPE & DISK TURBO
- PRINTERTOOL
- POWER RESET
- TOTAL BACKUP

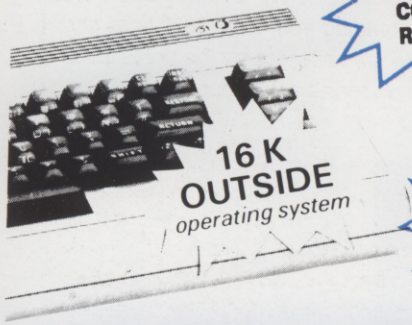
...."Money well spent" YC/CDU Jan 90

42 Pg Manual - "Damned Good Handbook" CCI Jan 90

SO MUCH FOR SO LITTLE

AVAILABLE FROM ALL GOOD COMPUTER RETAILERS

TRIED AND TESTED - OVER 100,000 SOLD IN EUROPE



...."highly recommended for C64 users" CCI Jan 90

YOU WILL WONDER HOW YOU EVER MANAGED WITHOUT IT

POWER TOOLKIT

A powerful BASIC-Toolkit (Additional helpful commands) that considerably simplifies programming and debugging.

AUTO	HARDCAT	RENUMBER
AUDIO	HARDCOPY	REPEAT
COLOR	HEX$	SAFE
DEEK	INFO	TRACE
DELETE	KEY	UNNEW
DOKE	PAUSE	QUIT
DUMP	PLIST	MONITOR
FIND	ILOAD	BLOAD

RENUMBER : *Also modifies all the GOTO's GOSUB's etc. Allows part of a program to be renumbered or displaced.*
PSET : *Set up of printer type.*
HARDCAT : *Prints out Directory.*

The toolkit commands can be used in your programs.

DISK TOOL

Using POWER CARTRIDGE you can load up to 6 times faster from disk.
The Disk commands can be used in your own programs.

DLOAD	DVERIFY	DIR
DSAVE	MERGE	DEVICE
DISK		

MERGE : *Two BASIC programs can be merged into one.*
DISK : *With DISK you can send commands directly to your disk.*

TAPE TOOL

Using POWER CARTRIDGE you can work up to 10 times faster with your data recorder. The Tape commands can be used in your own programs.

LOAD	SAVE	VERIFY
MERGE	AUDIO	

POWERMON

A powerful machine language monitor that is readily available and leaves all of your Commodore memory available for programming.
Also works in BASIC-ROM, KERNAL and I/O areas.

A ASSEMBLE	I INTERPRET	S SAVE
C COMPARE	J JUMP	T TRANSFER
D DIS-ASSEMBLE	L LOAD	V VERIFY
	M MEMORY	W WALK
F FILL	P PRINT	X EXIT
G GO	R REGISTER	$ DIRECTORY DOS Commands
H HUNT		

PRINTERTOOL

The POWER CARTRIDGE contains a very effective Printer-Interface, that self detects if a printer is connected to the Serial Bus or User-Port.
It will print all Commodore characters on Epson and compatible printers.
The printer-interface has a variety of set-up possibilities. It can produce HARDCOPY of screens not only on Serial printers (MPS801, 802, 803 etc) but also on Centronic printers (EPSON, STAR, CITIZEN, PANASONIC, etc).
The HARDCOPY function automatically distingishes between HIRES and LORES.
Multi-colour graphics are converted into shades of grey. The PSET functions allow you to decide on Large/Small and Normal/Inverse printing.
The printer PSET functions are:

PSET 0 - *Self detection Serial/Centronics.*
PSET 1 - *EPSON mode only.*
PSET 2 - *SMITH-CORONA mode only.*
PSET 3 - *Turns the printing 90 degrees!!*
PSET 4 - *HARDCOPY setting for MPS802/1526.*

PSET B - *Bit-image mode.*
PSET C - *Setting Lower/Upper case and sending Control Codes.*
PSET T - *All characters are printed in an unmodified state.*
PSET U - *Runs a Serial printer and leaves the User-port available.*
PSET Sx - *Sets the Secondary address for HARDCOPY with Serial Bus.*
PSET L1 - *Adds a line-feed, CHR$ (10), after every line.*
PSET L0 - *Switches PSET L1 off.*

Bitcon Devices Ltd does not authorise or purport to authorise the making by any means or for any purpose whatsoever of copies or adaptions of copyright works or other protected material, and users of the Power Cartridge must obtain the necessary prior consent for the making of such copies or adaptions from all copyright and other right owners concerned. See UK Copyright, Designs & Patents Act 1988.

POWER RESET

On the back of the POWER CARTRIDGE there is a Reset Button. Pressing this button makes a SPECIAL MENU appear on the screen.
This function will work with many programmes.

CONTINUE - *Allows you to return to your program.*
BASIC - *Return to BASIC.*
RESET - *Normal RESET.*
TOTAL BACKUP DISK - *Saves the contents of the memory onto a Disk. The program can be reloaded later with BLOAD followed by CONTINUE.*
RESET ALL TOTAL BACKUP TAPE - *RESET of any program. As BACKUP DISK but to TAPE.*
HARDCOPY - *At any moment, prints out a Hardcopy of the screen. Using CONTINUE afterwards you can return to the program.*
MONITOR - *Takes you into the Machine language Monitor.*

BDL

Bitcon Devices Ltd

88 BEWICK ROAD
GATESHEAD
TYNE AND WEAR
NE8 1RS
ENGLAND

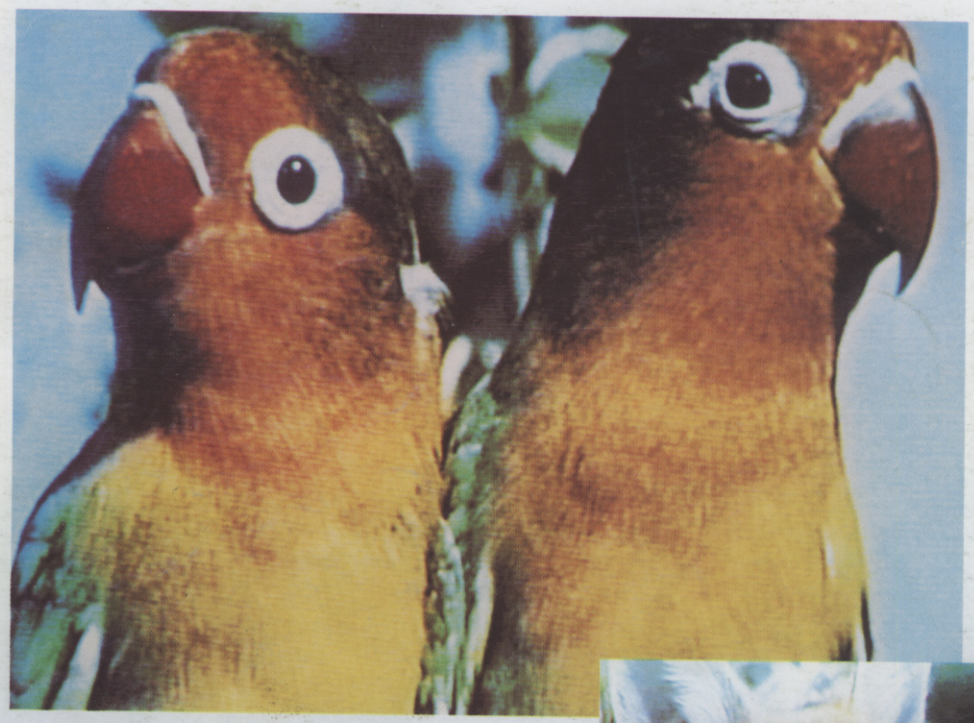
Tel: 091 490 1975 and 490 1919 Fax 091 490 1918
To order: Access/Visa welcome - Cheques or P/O payable to BDL
Price: £17.30 incl. VAT.
UK orders add £1.20 post/pack total - £18.50 incl. VAT.
Europe orders add £2.50. Overseas add £3.50
Scandinavian Mail Order and Trade enquiries to: Bhiab Elektronik, Box 216, Norrtalje 76123, SWEDEN. Tel: + 46 176 18425 Fax: 176 18401
TRADE AND EXPORT ENQUIRIES WELCOME

NEW

THE COMPLETE COLOUR SOLUTION

£179

Vidi ... No 1 in UK & Europe (Leading the way forward)



Actual unretouched digitised screenshot

Get the most out of your Amiga by adding:

"The Complete Colour Solution"

The Worlds ultimate creative leisure product for your Amiga. Capture dynamic high resolution images into your Amiga in less than one second.

And Look No Filters

Images can now be grabbed from either colour video camera, home VCR or in fact any still video source. The traditional method of holding three colour filters in front of your video camera is certainly a thing of the past. Because Vidi splits the RGB colours electronically there are no focussing or movement problems experienced by some of our slower competitors. Lighting is also less of an issue as light is not being shut out by lens filters. Put all this together with an already proven Vidi-Amiga/VidiChrome combination and achieve what is probably the most consistant and accurate high quality 4096 colour images ever seen on the Amiga.

The colour solution is fully compatible with all Amiga's from a standard A500 to the ultimate A3000. No additional RAM is required to get up and running.

You will see from independant review comments that we are undoubtedly their first choice and that was before the complete solution was launched. If you have just purchased your Amiga and are not sure what to buy next, then just read the comments or send for full review and demo disk.

Features ...

- Grab mono images from any video source
- Capture colour images from any still video source.
- Digitise up to 16 mono frames on a 1meg Amiga.
- Animate 16 shade images at different speeds.
- Create windows in both mono & colour.
- Cut & Paste areas from one frame to another.
- Hardware and software brightness & contrast control.
- Choice of capture resolutions standard & Dynamic interlace.
- Full Palette control.
- Add text or draw within art package.

Amiga Computing: The best Amiga digitiser has had the technicolour treatment. Vidi must be one of the most exciting peripherals you can buy for your Amiga.

Micro Mart: When I first saw Vidi "in the flesh" as it were, at the CES show last September it looked to be the answer to a frustrated Digi View owner's dreams – in fact to see pictures appearing on screen without the customary two minutes wait seemed almost too good to be true. I have consistently produced more good quality pictures in the short time I have had Vidi than I ever did with Digiview.

Zero: Now under normal circumstances cheap usually means poor quality but this is not the case with Rombo. Why? cos Vidi-Amiga is the best digitiser for under £500 and I've tried them all.

Amiga Format: Where quality is concerned, Vidi produces some of the best results I've seen on any digitiser at any price.

Amiga User International: The latest addition to the Rombokit is called Vidi-RGB and brings this already impressive package to the realms of totally amazing. CONCLUSION: Who will find Vidi-Amiga useful? The answer to this is almost anyone with a video recorder or camera and a passing interest in graphics.

Full colour demonstration disk available for only £1.95 to cover P&P.

6 Fairbairn Road, Livingston, EH54 6TS. Tel: 0506-414631 Fax: 0506-414634

Visit us at the 16 Bit Computer Show Novotel hotel, Hammersmith 12th-14th july 1991 Stand No. 101

ROMBO Limited